

EXHIBIT G

Exemplar Chart of U.S. Patent 7,411,582**U.S. Patent 5,638,501 (“Gough ‘501”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt ² and/or U.S. Patent No. 5,603,053 ³ and/or the MessagePad 120 ⁴ and/or U.S. Patent No. 5,946,499⁵
1	1. In a computing environment, a computer-implemented method comprising:	“This invention relates generally to computer systems, and more particularly to graphical user interfaces for computer systems.” Col. 1, ll. 5-7.

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Method and Apparatus for Displaying an Overlay Image” issued to Gough et al.; filed May 10, 1993; issued June 10, 1997.

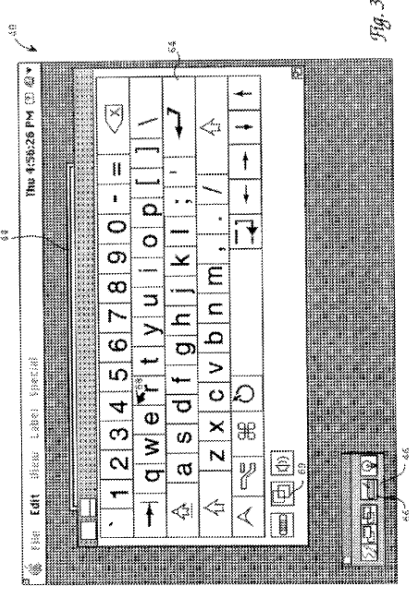
² Kraig Brockschmidt, Inside OLE (2d ed. 1995).

³ The Gough ‘501 patent incorporates by reference copending U.S. patent application Serial No. 08/060,438 (mistakenly listed as 08/060,458) entitled “Method and Apparatus for Interfacing With a Computer System.” Its inventors abandoned the ‘438 application in favor of a child application, U.S. patent application Serial No. 08/610,852, which they filed on March 5, 1996. The ‘852 application issued as U.S. Patent No. 5,603,053 (“Gough ‘053”) on February 11, 1997. The Gough ‘053 patent issued to Gough et al. and is entitled “System For Entering Data Into An Active Application Currently Running In The Foreground By Selecting An Input Icon In A Palette Representing Input Utility.”

⁴ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

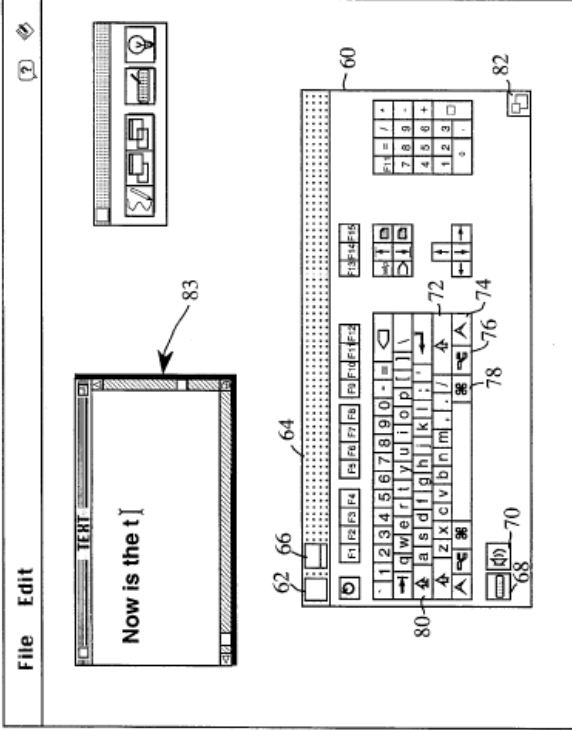
⁵ “Method And Apparatus For Processing Text Inputs From Multiple Input Devices In A Plurality Of Applications” issued to Saunders; filed May 10, 1996; issued August 31, 1999.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt ² and/or U.S. Patent No. 5,603,053 ³ and/or the MessagePad 120 ⁴ and/or U.S. Patent No. 5,946,499 ⁵
1.1	displaying an actuatable icon representative of an input method list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a computer-executable software component distinct from the computer programs;	<p>The invention provides for multiple types of overlays to be used as inputs. Col. 7, ll. 60-63 (“Of course, other overlay images besides keyboards can be provided by the present invention, e.g. handwriting “recognition” windows, etc.”).</p> <p>These overlays are executable independent of the application programs. Col. 2, ll. 29-33 (“Preferably, the base image is produced by an unmodified application program running on the computer system, and the overlay image is produced by a computer implemented process of the present invention referred to herein as the “overlay utility”.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The Gough ‘053 describes a method for “displaying a palette having at least one icon representing an input utility, and selecting the icon to create an input image.” Gough ‘053 Col. 3, ll. 15-17. In one embodiment of the Gough ‘053 patent, the palette can be set to launch itself “upon the start-up of the computer system[.]” Gough ‘053 Col. 6, ll. 12-13.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to display an actuatable icon representative of an input method list to the user. For example, Gough ‘501 displays palette 46, and it would have been obvious to one of skill in the art to display palette 46 in response to a user request. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the display of an actuatable icon representative of an input method list to the user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to</p>

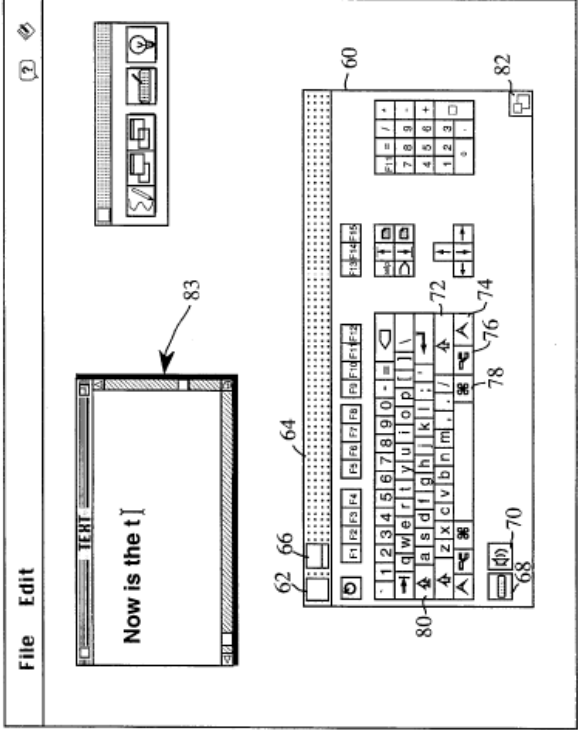
Claim	U.S. Patent 5,638,501 (“Gough ‘501’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt ² and/or U.S. Patent No. 5,603,053 ³ and/or the MessagePad 120 ⁴ and/or U.S. Patent No. 5,946,499 ⁵
	combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.2	<p>in response to actuation of the actuatable icon, displaying the input method list;</p> <p>See Claim 1.1 regarding the actuation of the actuatable icon.</p> <p>Regarding the display of the input method list, these input methods are shown by the utility palette.</p> <p>See Figure 3.</p>  <p>The Gough ‘053 describes a method for “displaying a palette having at least one icon representing an input utility, and selecting the icon to create an input image.” Gough ‘053 Col. 3, ll. 15-17. In one embodiment of the Gough ‘053 patent, the palette can be set to launch itself “upon the start-up of the computer system[.]” Gough ‘053 Col. 6, ll. 12-13.</p>
1.3	<p>receiving a selection of an input method from the input method list;</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”). Figures 4a and 4b of the Gough ‘053 patent demonstrate the selection (by dragging) of</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt² and/or U.S. Patent No. 5,603,053³ and/or the MessagePad 120⁴ and/or U.S. Patent No. 5,946,499⁵</p>
		<p>the keyboard image from the input method list.</p> <div data-bbox="380 518 709 934"> </div> <p style="text-align: center;"><i>Figure 4a</i></p> <div data-bbox="766 493 1133 961"> </div> <p style="text-align: center;"><i>Figure 4c</i></p>
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	<p>“Preferably, this keyboard image 64 is provided by dragging a keyboard icon 66 off of the PenBoard palette 46 in a fashion more fully described in copending U.S. patent application Ser. No. 08/060,458, filed May 10, 1993, on behalf of Gough et al., entitled “Method and Apparatus for Interfacing With a Computer System”, and assigned to the assignee of the present application, the disclosure of which is hereby</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt² and/or U.S. Patent No. 5,603,053³ and/or the MessagePad 120⁴ and/or U.S. Patent No. 5,946,499⁵</p> <p>incorporated herein by reference in its entirety.” Col. 6, ll. 18-26.</p> <p>Once installed, the input method may communicate with the active application. See Claim 1 of Gough ‘501 (“[I]n response to said translucency icon being selected, translucently displaying said overlay image[.]”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Claim 1 of Gough ‘053 (“A method for launching an input utility . . . said method comprising . . . activating said input utility and displaying an input image created for said input utility at about said desired location, said input image dedicating a portion of said screen for data input into said input utility; and linking said input image to said active application such that data input into said input utility is communicated to said active application.”).</p>
1.5	receiving input via the interactive input panel; and	<p>“Information can be entered into the window 44 of the application program from the keyboard image 64 by ‘tapping’ on a ‘key’ with the stylus 38. For example, arrow 68 on the keyboard image 64 represents the “tapping” on the key “R” with the stylus 38. This tapping action will send a “R” to be displayed in the window 44 of the AppleShare application just as if a “R” had been typed on a physical keyboard.” Col. 6, ll. 30-37.</p> <p>See Claim 3 of Gough ‘501 (“A method as recited in claim 2 wherein said second computer implemented process intercepts screen inputs which contact said overlay image and processes said screen inputs.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Claim 3 of Gough ‘053 (“A method for inputting data to an active application . . . the method comprising . . . detecting the engagement of said input image by a pointer means; analyzing said engagement to determine input data; and sending said input</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt² and/or U.S. Patent No. 5,603,053³ and/or the MessagePad 120⁴ and/or U.S. Patent No. 5,946,499⁵</p> <p>data to said active application program.”).</p> <p>See Figure 4c of Gough ‘053, which shows active application receiving input data from the input panel.</p>
		 <p style="text-align: center;"><i>Figure 4c</i></p>
1.6	providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.	<p>“This tapping action will send a “R” to be displayed in the window 44 of the AppleShare application just as if a “R” had been typed on a physical keyboard.” Col. 6, ll. 34-37.</p> <p>See Claim 6 of Gough ‘501 (“A method as recited in claim 3 wherein said second computer implemented process is further operative to update said first computer implemented process according to the step of processing said screen inputs.”).</p> <p>The prior art includes input windows that accept user input and transmit that input to programs “as if it were typed from a keyboard.” Col. 2, ll. 5-8. The improvement of</p>

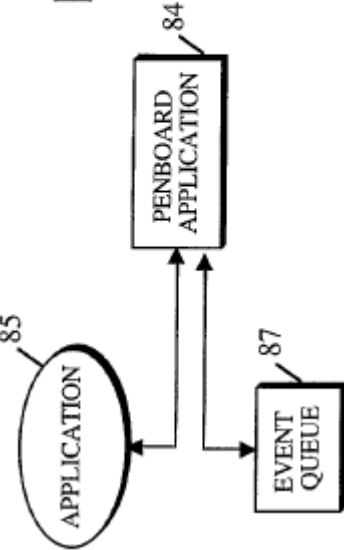
Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt² and/or U.S. Patent No. 5,603,053³ and/or the MessagePad 120⁴ and/or U.S. Patent No. 5,946,499⁵</p> <p>this invention is that the input window is transparent. Col. 2, ll. 15-17 (“The overlay image can serve as an input device for application programs without obscuring images made on the screen by the application programs.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Claim 3 of Gough ‘053 (“A method for inputting data to an active application . . . the method comprising . . . detecting the engagement of said input image by a pointer means; analyzing said engagement to determine input data; and sending said input data to said active application program.”).</p> <p>See Figure 4c of Gough ‘053, which shows active application receiving input data from the input panel.</p>
-------	-----------------------	--

Claim	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt ² and/or U.S. Patent No. 5,603,053 ³ and/or the MessagePad 120 ⁴ and/or U.S. Patent No. 5,946,499 ⁵
	 <p style="text-align: center;"><i>Figure 4c</i></p>

Claim	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
2	<p>See Figures 3-5.</p> <p>“In FIG. 2, a screen 40 of a Macintosh computer system made by Apple Computer, Inc., of Cupertino, Calif., includes a desktop image 42 produced by a Macintosh operating system[.]” Col. 5, ll. 48-51.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		Compare Col. 8, ll. 26-30 (“Running under the operating system 96 is an application program 98, such as the aforementioned AppleShare application program. Application program 98, when it wants to open a window such as window 44 of FIG. 2, calls a set of routines 100 provided by the operating system 96.”) with Col. 7, ll. 1-5 (“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program and be displayed in a password field of the window 44.”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
3	3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.	<p>“It will be apparent with a study of FIGS. 4 and 5a-5c that the translucent keyboard image 64’ is a distinctly superior user <i>interface</i> for situations in which screen area is at a premium. Since images “beneath” the translucent keyboard image 64’ can be seen through the keyboard image, the user has immediate feedback as to the accuracy of his or her input to the active application program.” Col. 8, ll. 23-26 (emphasis added).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Figure 5 of Gough ‘053.</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499</p>
		<p style="text-align: center;">Figure 5</p>  <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system capable of calling functions in the manager component via a defined interface set. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and U.S. Patent 5,157,384 (“Greanias ‘384”) renders this claim obvious. Greanias ‘384 teaches calling functions in the manager component via a defined interface set. See Greanias ‘384 Claim 37 and Col. 8, ll. 59-65. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
4	4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same manner as if the input was received via a hardware keyboard.	<p>“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program and be displayed in a password field of the window 44.” Col. 7, ll. 1-5.</p> <p>The prior art includes input windows that accept user input and transmit that input to programs “as if it were typed from a keyboard.” Col. 2, ll. 5-8.</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55.</p> <p>The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Col. 7, ll. 39-44 of Gough ‘053 (“[T]he computer implemented process of the present invention intercepts calls made by TeachText to the input event queue and passes characters to TeachText from the keyboard image 60 just as if they had been entered into the computer system from a physical keyboard.”).</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving information corresponding to a keyboard character entered via the displayed keyboard.	<p>“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program and be displayed in a password field of the window 44.” Col. 7, ll. 1-5.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
8	8. The method of claim 1 further comprising, hiding the input panel.	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499</p> <p>See Figures 3 & 6b.</p> <p>“The process 80 begins at 84, and in a step 86, it is determined whether the process 80 is completed. In this instance, <i>a process 80 is completed when the “button” 69 of the translucent keyboard image 64’ (see FIG. 4) is tapped</i>. If the process is completed, the overlay utility 80 is <i>terminated</i> as indicated at 88. If the process is not completed, a step 90 displays an “overlay” image on the screen such that images on the screen that it overlaps can be seen through the overlay image.” Col. 7, ll. 50-58 (emphasis added).</p> <p>See Claim 20 of Gough ‘501 (“A pen computer system comprising . . . means coupled to said screen means for displaying an overlay image . . . said means for displaying said overlay image responsive to a users selection to toggle between (i) opaquely displaying said overlay image and (ii) translucently displaying said overlay image.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”). The Gough ‘053 patent permits utility input windows to be “hidden from view when not required.” Col. 3, ll. 48 of Gough ‘053.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
9	9. The method of claim 1 further comprising, docking the input panel.	<p>“[T]he palette 46 (<i>which is just a specialized form of window</i>) is produced by the PenBoard application, and does not occupy the entire space of the screen 40.” Col. 6, ll. 4-7 (emphasis added).</p> <p>See Claim 8 of Gough ‘501 (“A method as recited in claim 1 wherein said computer system is a pointer based computer system, said first utility icon is selected by dragging said first utility icon off of said utility palette with said pointer, and said overlay image is displayed in a location correlated to where said first utility icon is dragged.”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

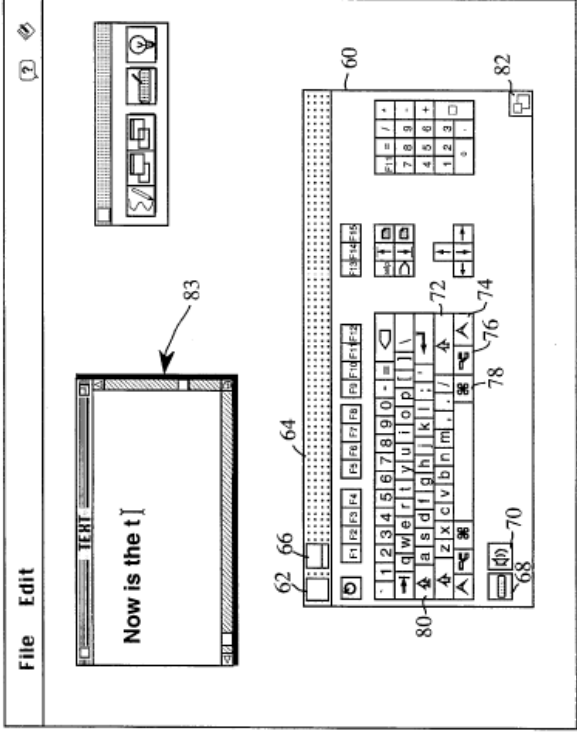
Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
10	10. At least one computer-readable medium having computer-executable instructions, which when executed perform the method of claim 1.	<p>“As shown in FIG. 1, a pen computer system 10 in accordance with the present invention includes a central processing unit (CPU) 12, read only memory (ROM) 14, random access memory (RAM) 16, expansion RAM 17, input/output (I/O) circuitry 18, display assembly 20, and expansion bus 22. The pen computer system 10 may also optionally include a mass storage unit 24 such as a disk drive unit or nonvolatile memory such as flash memory and a real-time clock 26.” Col. 4, ll. 32-40.</p> <p>“A pen computer system in accordance with the present invention includes a central processing unit (CPU), a screen assembly coupled to the CPU, a mechanism coupled to the screen assembly for displaying a base image on the screen assembly, and a mechanism coupled to the screen assembly for displaying an overlay image on the screen assembly such that portions of the base image which are overlapped by the overlay image are at least partially visible through the overlay image.” Col. 2, ll. 54-62.</p>

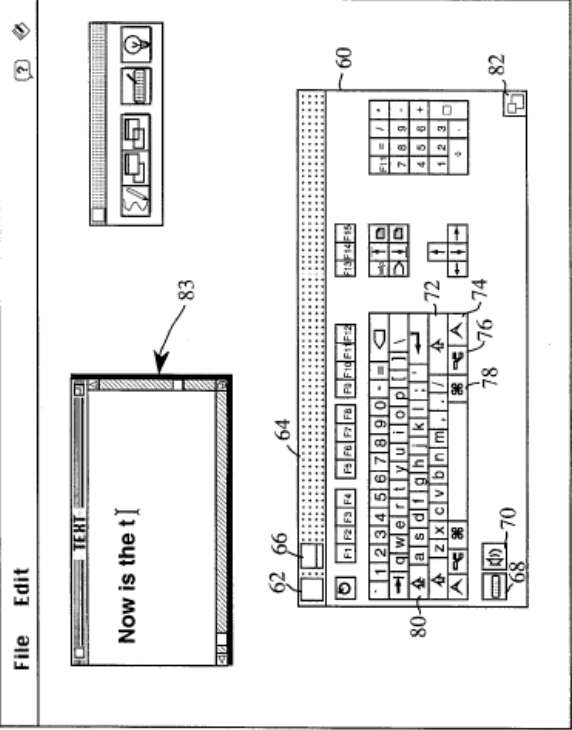
Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	<p>“As shown in FIG. 1, a pen computer system 10 in accordance with the present invention includes a central processing unit (CPU) 12, read only memory (ROM) 14, random access memory (RAM) 16, expansion RAM 17, input/output (I/O) circuitry 18, display assembly 20, and expansion bus 22. The pen computer system 10 may also optionally include a mass storage unit 24 such as a disk drive unit or nonvolatile memory such as flash memory and a real-time clock 26.” Col. 4, ll. 32-40.</p> <p>“A pen computer system in accordance with the present invention includes a central processing unit (CPU), a screen assembly coupled to the CPU, a mechanism coupled to the screen assembly for displaying a base image on the screen assembly, and a</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		mechanism coupled to the screen assembly for displaying an overlay image on the screen assembly such that portions of the base image which are overlapped by the overlay image are at least partially visible through the overlay image.” Col. 2, ll. 54-62.
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each executable input method is an interchangeable software component distinct from one or more application programs, each executable input method having a defined interface set such that the executable input method is connectable to the application programs;	<p>The invention provides for multiple types of overlays to be used as inputs. Col. 7, ll. 60-63 (“Of course, other overlay images besides keyboards can be provided by the present invention, e.g. handwriting “recognition” windows, etc.”).</p> <p>These overlays can be used with multiple programs. Col. 3, ll. 23-28 (“Another advantage of the overlay image of the present invention is that it works with both pen-aware and non-pen-aware application programs. Therefore, the overlay image of the present invention can be used with the many thousands of application programs which are not designed to be used in pen computer systems.”).</p> <p>These overlays are executable independent of the application programs. Col. 2, ll. 29-33 (“Preferably, the base image is produced by an unmodified application program running on the computer system, and the overlay image is produced by a computer implemented process of the present invention referred to herein as the “overlay utility”.”).</p> <p>The overlays are connectable to the application programs. Col. 2, ll. 42-47 (“Preferably, the step of running the overlay program includes the steps of. . . 2) intercepting screen inputs which contact the overlay image; 3) processing the intercepted screen inputs in the CPU; and 4) updating the application program based upon the process screen inputs.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>Figures 4a and 4b of the Gough ‘053 patent demonstrate the selection (by dragging) of the keyboard image from the input method list.</p>

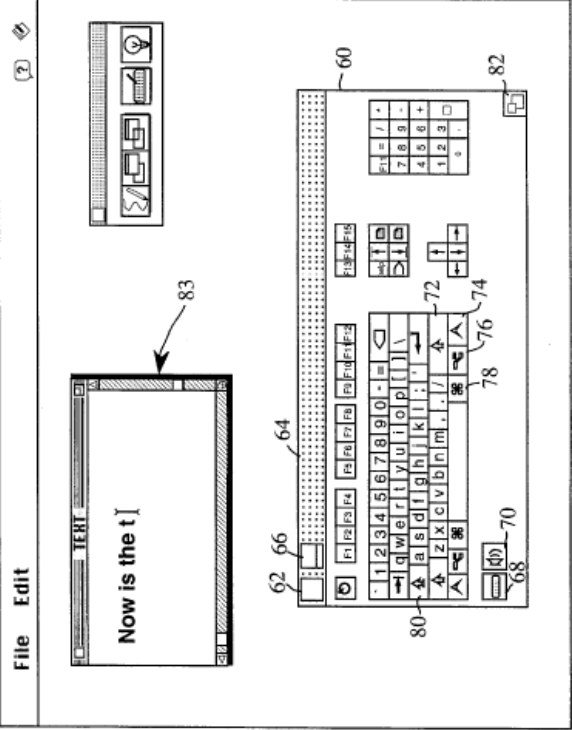
Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499</p>
		<div data-bbox="329 520 657 934"> </div> <p data-bbox="669 682 690 756"><i>Figure 4a</i></p> <div data-bbox="716 493 1083 961"> </div> <p data-bbox="1094 682 1115 756"><i>Figure 4c</i></p> <p data-bbox="1144 170 1250 1289">To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system capable of calling functions in the manager component via a defined interface set. Col. 14, ll. 23-29.</p> <p data-bbox="1271 170 1451 1289">In addition, the combination of Gough ‘501 and U.S. Patent 5,157,384 (“Greanias ‘384”) renders this claim obvious. Greanias ‘384 teaches calling functions in the manager component via a defined interface set. See Greanias ‘384 Claim 37 and Col. 8, ll. 59-65. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
11.2	opening an input window on a display of the computer system independent of a window of an active application program; and	The overlay utility opens an input window. Col. 2, ll. 42-44 (“Preferably, the step of running the overlay program includes the step[] of . . . displaying an overlay image on the screen[.]”). These overlays are executable independent of the application programs. Col. 2, ll. 29-33 (“Preferably, the base image is produced by an unmodified application program running on the computer system, and the overlay image is produced by a computer implemented process of the present invention referred to herein as the “overlay utility”:”). The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”). See Figure 4c of Gough ‘053, which shows independent windows for the palette and keyboard input window.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
11.3	displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input received by the selected executable input method via the interactive input panel is provided to the active application program as if the information was received via user input at a hardware input device.	 <p style="text-align: center;"><i>Figure 4c</i></p>
		<p>See Figures 3-5.</p> <p>See Col. 6, ll. 26-37 discussing the prior art opaque overlays (“As can be seen in this FIG. 3, the keyboard image 64 completely obscures the icons 52, 54 and 56 of FIG. 2, and almost totally obscures the window 44 of the AppleShare application program. Information can be entered into the window 44 of the application program from the keyboard image 64 by “tapping” on a “key” with the stylus 38. For example, arrow 68 on the keyboard image 64 represents the “tapping” on the key “R” with the stylus 38. This tapping action will send a “R” to be displayed in the window 44 of the AppleShare application just as if a “R” had been typed on a physical keyboard.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55.</p>

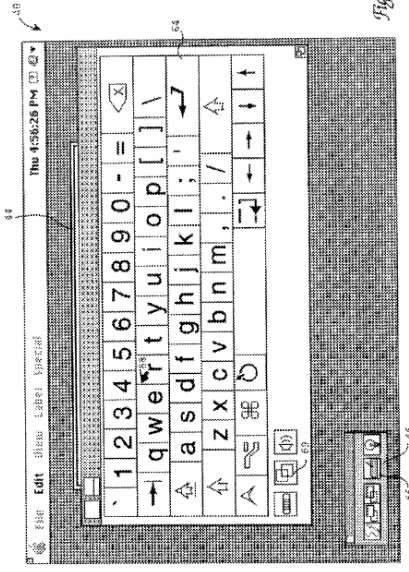
Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499</p>
		<p>The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Claim 3 of Gough ‘053 (“A method for inputting data to an active application . . . the method comprising . . . detecting the engagement of said input image by a pointer means; analyzing said engagement to determine input data; and sending said input data to said active application program.”).</p> <p>See Figure 4c of Gough ‘053, which shows: (1) independent windows for the palette and keyboard input window; and (2) an active application receiving input data from the input panel.</p>
		 <p style="text-align: center;"><i>Figure 4c</i></p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
13	13. The computer-readable medium of claim 11 further comprising, providing a Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuable to display a selectable list of the plurality of executable input methods.	<p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The Gough ‘053 describes a method for “displaying a palette having at least one icon representing an input utility, and selecting the icon to create an input image.” Gough ‘053 Col. 3, ll. 15-17. In one embodiment of the Gough ‘053 patent, the palette can be set to launch itself “upon the start-up of the computer system[.]” Gough ‘053 Col. 6, ll. 12-13.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to display an actuable icon representative of an input method list to the user. For example, Gough ‘501 displays palette 46, and it would have been obvious to one of skill in the art to display palette 46 in response to a user request. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the display of an actuable icon representative of an input method list to the user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
14	14. The computer-readable medium of claim 13 further comprising, receiving a selection of one of the plurality of executable input methods displayed in the list as a selected executable input method, and in response, closing any open input window, and opening a new input window corresponding to the selected executable input method.	<p>The overlay utility opens an input window. Col. 2, ll. 42-44 (“Preferably, the step of running the overlay program includes the step[] of . . . displaying an overlay image on the screen[.]”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Figure 4c of Gough ‘053, which shows independent windows for the palette and keyboard input window.</p>  <p><i>Figure 4c</i></p> <p>The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”). The Gough ‘053 patent permits utility input windows to be “hidden from view when not required.” Col. 3, ll. 48 of Gough ‘053.</p>

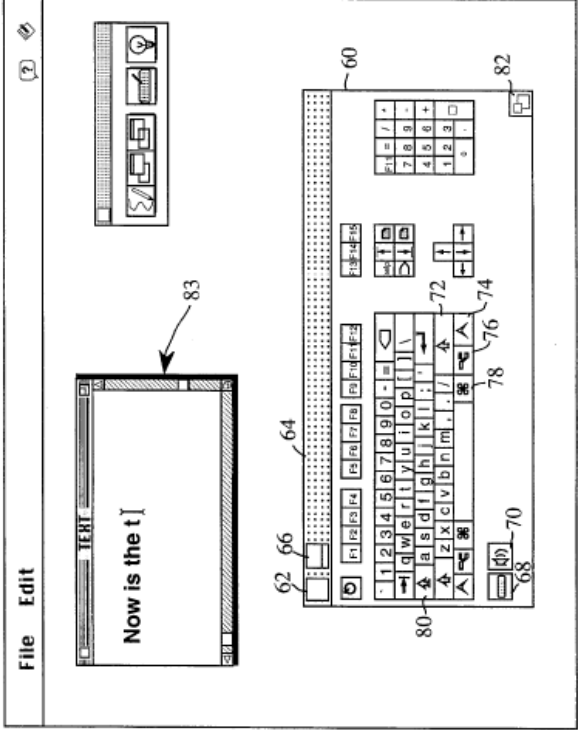
Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and U.S. Patent No. 5,148,155 (Martin ‘155) renders this claim obvious. Martin ‘155 teaches the hiding of an input panel. See Martin ‘155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
15	15. At least one computer-readable medium having computer-executable instructions, which when executed perform steps, comprising:	<p>“As shown in FIG. 1, a pen computer system 10 in accordance with the present invention includes a central processing unit (CPU) 12, read only memory (ROM) 14, random access memory (RAM) 16, expansion RAM 17, input/output (I/O) circuitry 18, display assembly 20, and expansion bus 22. The pen computer system 10 may also optionally include a mass storage unit 24 such as a disk drive unit or nonvolatile memory such as flash memory and a real-time clock 26.” Col. 4, ll. 32-40.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	<p>The invention provides for multiple types of overlays to be used as inputs. Col. 7, ll. 60-63 (“Of course, other overlay images besides keyboards can be provided by the present invention, e.g. handwriting “recognition” windows, etc.”).</p> <p>These overlays are executable independent of the application programs. Col. 2, ll. 29-33 (“Preferably, the base image is produced by an unmodified application program running on the computer system, and the overlay image is produced by a computer implemented process of the present invention referred to herein as the “overlay utility”.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55.</p> <p>Figure 3 of the Gough ‘501 patent shows the utility palette of the invention which presents icons representative of the various input methods.</p>  <p>To the extent not expressly or inherently disclosed, it would have been obvious to display the input method list in response to actuation of the actuatable icon. For example, Gough ‘501 displays palette 46, and it would have been obvious to one of skill in the art to display palette 46 in response to a user request. Col. 9, ll. 64-67 to Col. 10, ll. 1-3 of Gough ‘053.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
15.2	invoking a selected input method in response to a user selecting an icon corresponding to the selected input method, including presenting an input panel window; and	<p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>Figures 4a and 4b of the Gough ‘053 patent demonstrate the selection (by dragging) of the keyboard image from the input method list, and the presentation of the keyboard input panel window.</p> <div data-bbox="578 520 906 934"> </div> <p style="text-align: center;"><i>Figure 4a</i></p> <div data-bbox="963 495 1333 963"> </div> <p style="text-align: center;"><i>Figure 4c</i></p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	<p>“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program and be displayed in a password field of the window 44.” Col. 7, ll. 1-5.</p> <p>See Claim 3 of Gough ‘501 (“A method as recited in claim 2 wherein said second computer implemented process intercepts screen inputs which contact said overlay image and processes said screen inputs.”).</p> <p>See Claim 6 of Gough ‘501 (“A method as recited in claim 3 wherein said second computer implemented process is further operative to update said first computer implemented process according to the step of processing said screen inputs.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Claim 3 of Gough ‘053 (“A method for inputting data to an active application . . . the method comprising . . . detecting the engagement of said input image by a pointer means; analyzing said engagement to determine input data; and sending said input data to said active application program.”).</p> <p>See Figure 4c of Gough ‘053, which shows active application receiving input data from the input panel as if typed from a normal keyboard.</p>

Claim	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
	 <p style="text-align: center;"><i>Figure 4c</i></p>

Claim	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
16	<p>16. The computer-readable medium of claim 15 wherein accepting user data includes detecting user interaction with a touch-sensitive display.</p> <p>“A method and apparatus for providing a translucent overlay image over a base image on the screen of a computer system.” Abstract.</p> <p>Like the prior art, the transparent overlay keyboard “provide[s] a recognition window for inputting handwritten data which is then recognized and sent to an application</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		program as if it were typed from a keyboard[.]” Col. 2, ll. 6-8.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
17	17. The computer-readable medium of claim 15 wherein each input method comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 14, ll. 23-29. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Gough ‘501 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		<p>common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p><u>The combination of Gough ‘501 and U.S. Patent No. 5,946,499 (“the Saunders Reference”) also renders this claim obvious. In the Saunders Reference, text input methods communicate with the text services manager (“TSM”) through an object model interface. See Saunders Reference, Col. 4, ll. 23-43 (“Each second data structure TSMContext 40 is suitably attached to the first data structure TSMDocument 41 and contains two pieces of information, a LocaleObject 42 and a SOM (system object model) object 44. A LocaleObject 42 appropriately represents a text service and supplies the TSM 20 with the information necessary to load that text service, including a text service class used by the TSM 20 to access the text service and a SOM class name for the SOM class implemented by the text service, as is well understood by those skilled in the art. A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</u></p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified,</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and U.S. Patent No. 5,454,046 (“the Carmen Reference”)⁶ renders this claim obvious. See Carmen Reference, Col. 10, ll. 14-22 (“The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation defined by the Unicode Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all languages of the world, whereas ASCII and ANSI representations are limited to 256 characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p>

⁶ “Universal Symbolic Handwriting Recognition System” issued to Carman; filed September 17, 1993; issued September 26, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		<p>See Figure 3A of Carmen Reference.</p> <div data-bbox="428 478 857 989" data-label="Diagram"> </div> <p>FIG. 3A</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>Similarly, the combination of Gough ‘501 and U.S. Patent No. 5,455,901 (“the Friend Reference”)⁷ renders this claim obvious. See Claim 6 of Friend Reference (“The device of claim 5 wherein said translator comprises a generator for generating machine readable characters from said strokes.”); Claim 7 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise Unicode</p>

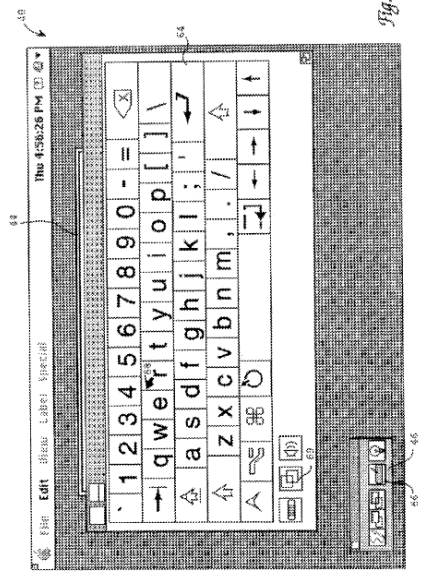
⁷ “Input Device With Deferred Translation” issued to Friend; filed September 12, 1994; issued October 3, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		characters.”); Claim 8 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise ASCII characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
19	19. In a computing environment, a system comprising,	“This invention relates generally to computer systems, and more particularly to graphical user interfaces for computer systems.” Col. 1, ll. 5-7.
19.1	a manager component stored on one or more computer-readable media and configured: to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable	The input method utility program serves as the manager component to interface between the user input and the application [here, Appleshare] and send the user input to the application window. Col. 7, ll. 1-5 (“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program [the open application] and be displayed in a password field of the window 44.”). See Figure 6b. See also Col. 7, ll. 63-65 (“Next, in a step 92, the overlay utility intercepts screen inputs which contact the overlay image, and these screen inputs are

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
	software component distinct from one or more computer programs, and to send input data corresponding to a user input received at the selected input method to a graphical windowing environment; and	processed.”). The invention provides for multiple types of overlays to be used as inputs. Col. 7, ll. 60-63. These overlays can be used with multiple programs. Col. 3, ll. 23-28. These overlays are executable independent of the application programs. Col. 2, ll. 29-33.
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	The application receives the input data – in its application window – from the utility application as if were entered on a keyboard. See Claim 15 of Gough ‘501 (A method as recited in claim 14 wherein said step of running an overlay program comprises the steps of: intercepting screen inputs which contact said translucent overlay image; processing said intercepted screen inputs in said CPU; and updating said application program based upon said processed screen inputs.”); see also Col. 2, ll. 6-8 (Like the prior art, the transparent overlay keyboard “provide[s] a recognition window for inputting handwritten data which is then recognized and sent to an application program as if it were typed from a keyboard[.]”).
Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	“[T]he keyboard image 64 can be used to input data into a currently active application program (such as AppleShare)[.]” Col. 6, ll. 40-42.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 ("Gough '501") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
21	21. The system of claim 19 further comprising an input panel window corresponding to the selected input method.	See Figures 3-5. "The present invention provides a transparent overlay image over a base image provided on a screen of a pen computer system. The overlay image can serve as an input device for application programs without obscuring images made on the screen by the application programs." Col. 6, ll. 12-16.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 ("Gough '501") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	See Claim 7 of Gough '501 ("A method as recited in claim 3 wherein said overlay image is a keyboard image comprising icons which represent alphanumeric characters"). See Figure 3.
		 <p>Fig. 3</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
23	23. The system of claim 21 wherein the manager component selectively displays and hides the input panel window.	<p>The overlay utility opens an input window. Col. 2, ll. 42-44 (“Preferably, the step of running the overlay program includes the step[] of . . . displaying an overlay image on the screen[.]”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55.</p> <p>The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Figure 4c of Gough ‘053, which shows independent windows for the palette and keyboard input window.</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499</p>
		<div data-bbox="342 365 915 1094"> </div> <p style="text-align: center;"><i>Figure 4c</i></p> <p>The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”). The Gough ‘053 patent permits utility input windows to be “hidden from view when not required.” Col. 3, ll. 48 of Gough ‘053.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and U.S. Patent No. 5,148,155 (Martin ‘155) renders this claim obvious. Martin ‘155 teaches the hiding of an input panel. See Martin ‘155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period.
		Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
25	25. The system of claim 19 where the input method is displayed on a touch-sensitive display screen.	“A method and apparatus for providing a translucent overlay image over a base image on the screen of a computer system.” Abstract. Like the prior art, the transparent overlay keyboard “provide[s] a recognition window for inputting handwritten data which is then recognized and sent to an application program as if it were typed from a keyboard[.]”. Col. 2, ll. 6-8.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
26	26. The system of claim 19 wherein the manager component transfers	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design an input manager capable of transferring information from

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
	information from the computer program to the selected input method.	the computer program to the selected input method. Col. 14, ll. 23-29. In addition, the combination of Gough ‘501 and U.S. Patent No. 5,148,155 (Martin ‘155) renders this claim obvious. Martin ‘155 teaches the transferring information from the computer program to the selected input method. See Martin ‘155 Col. 32, ll. 36-44. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined interface set.	The input method utility program serves as the manager component to interface between the user input and the application [here, Appleshare] and send the user input to the application window. Col 7, ll. 1-5 (“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program [the open application] and be displayed in a password field of the window 44.”). To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system capable of calling functions in the manager

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		component via a defined interface set. Col. 14, ll. 23-29. In addition, the combination of Gough ‘501 and U.S. Patent 5,157,384 (“Greanias ‘384”) renders this claim obvious. Greanias ‘384 teaches calling functions in the manager component via a defined interface set. See Greanias ‘384 Claim 37 and Col. 8, ll. 59-65. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
28	28. The system of claim 19 wherein the selected input method comprises an object.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 14, ll. 23-29. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499</p>
		<p>necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Gough ‘501 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>The combination of Gough ‘501 and U.S. Patent No. 5,946,499 (“the Saunders Reference”) also renders this claim obvious. In the Saunders Reference, text input methods communicate with the text services manager (“TSM”) through an object model interface. See Saunders Reference, Col. 4, ll. 23-43 (“Each second data structure TSMContext 40 is suitably attached to the first data structure TSMDocument 41 and contains two pieces of information, a LocaleObject 42 and a SOM (system object model) object 44. A LocaleObject 42 appropriately represents a text service and supplies the TSM 20 with the information necessary to load that text service, including a text service class used by the TSM 20 to access the text service and a SOM class name for the SOM class implemented by the text service, as is well understood by those skilled in the art. A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		<u>described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</u> Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
29	29. The system of claim 19 wherein the selected input method draws an input panel in an input panel window displayed in the graphical windowing environment.	See Figures 3-5.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
30	30. The system of claim 29 wherein the manager component selectively displays and hides the display of the input panel window.	<p>The overlay utility opens an input window. Col. 2, ll. 42-44 (“Preferably, the step of running the overlay program includes the step[] of . . . displaying an overlay image on the screen[.]”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Figure 4c of Gough ‘053, which shows independent windows for the palette and keyboard input window.</p> <div data-bbox="683 367 1258 1094"> </div> <p style="text-align: center;"><i>Figure 4c</i></p> <p>The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”). The Gough ‘053 patent permits utility input windows to be “hidden from view when not required.” Col. 3, ll. 48 of Gough ‘053.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and U.S. Patent No. 5,148,155 (Martin ‘155) renders this claim obvious. Martin ‘155 teaches the hiding of an input panel. See Martin ‘155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
31	31. The system of claim 29 wherein the manager component docks the input panel window.	<p>“[T]he palette 46 (<i>which is just a specialized form of window</i>) is produced by the PenBoard application, and does not occupy the entire space of the screen 40.” Col. 6, ll. 4-7 (emphasis added).</p> <p>See Claim 8 of Gough ‘501 (“A method as recited in claim 1 wherein said computer system is a pointer based computer system, said first utility icon is selected by dragging said first utility icon off of said utility palette with said pointer, and said</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499</p> <p>overlay image is displayed in a location correlated to where said first utility icon is dragged.”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 9, ll. 64-67 to Col. 10, ll. 1-3 of Gough ‘053.</p> <p>In addition, the combination of Gough ‘501 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
-------	-----------------------	--

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
41	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	See claim 10.
42	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.

EXHIBIT H

Exemplar Chart of U.S. Patent 7,411,582**U.S. Patent 5,157,384 (“Greanias ‘384”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499 ⁴
1	1. In a computing environment, a computer-implemented method comprising:	“An advanced user interface for use with a computer system operating on an integrated operating environment.” Abstract.
1.1	displaying an actuatable icon representative of an input method list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a computer-executable software component distinct from the computer	“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract. “It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 30-32. “A few examples of utilities which might be found in the advanced user interface

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Advanced User Interface” issued to Greanias et al.; filed April 28, 1989; issued October 20, 1992.

² Kraig Brockschmidt, Inside OLE (2d ed. 1995).

³ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

⁴ “Method And Apparatus For Processing Text Inputs From Multiple Input Devices In A Plurality Of Applications” issued to Saunders; filed May 10, 1996; issued August 31, 1999.

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt² and/or the MessagePad 120³ and/or U.S. Patent No. 5,946,499⁴</p> <p>utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 14-23 (emphasis added).</p> <p>The input methods are each separately executable from the application programs. Col. 7, ll. 25 (“the AUI is written as an application program”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement an actuable icon representative of an input method list. For example, the invention discloses several types of interface utilities 109, and it would have been obvious to one of skill in the art to include this step to permit the user to select one of those input utilities. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p> <p>In addition, the combination of Greanias ‘384 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the presentation of an actuable icon representative of an input method list to a user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>See Claim 1.1 regarding the actuation of the actuable icon.</p> <p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract.</p>
1.2	in response to actuation of the actuable icon, displaying the input method list;	

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt² and/or the MessagePad 120³ and/or U.S. Patent No. 5,946,499⁴</p> <p>“It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 30-32.</p> <p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to present an input list to a user. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p> <p>In addition, the combination of Greanias ‘384 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the presentation of an input list to a user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.3	receiving a selection of an input method from the input method list;	See Claim 1.2.
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	<p>A user can install an input method by using the appropriate method. Col. 8, ll. 20-23. (“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.”).</p> <p>Once installed, the input method can use the translation capabilities of the AUI 100 to communicate with the active application. Col. 7, ll. 15. See also Col. 8, ll. 23-30 (“A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard. The space needed for the keyboard or keypad is quite small, and will not completely obscure the work area of the spreadsheet, word processor, etc., which is the active application program.”).</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499 ⁴
1.5	receiving input via the interactive input panel; and	Once installed, the input method can use the translation capabilities of the AUI 100 to communicate with the active application. Col. 7, ll. 15. See also Col. 8, ll. 23-30 (“A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard. The space needed for the keyboard or keypad is quite small, and will not completely obscure the work area of the spreadsheet, word processor, etc., which is the active application program.”).
1.6	providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.	<p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code. The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface.” Abstract.</p> <p>“The alternative input subsystem module provides communication between the attached user friendly input devices and the remainder of the advanced user interface as well as application programs through the integrated operating environment. The alternative input subsystem module receives the input signals generated by the input devices and translates them to input messages useable in the advanced user interface.” Col. 3, ll. 57-65.</p> <p>The AUI manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs. Col. 7, ll. 15.</p> <p>“After receiving an input message from Presentation Manager.TM. 156 together with information about which is the active application program, the PM-Link 201 refers back to the alternate input subsystem 203 to determine whether the “keyboard” or “mouse” message it received is in fact a keyboard or a mouse message, or rather a touch, voice, image, or other message. The PM-Link 201 then passes the true message to the appropriate application program.” Col. 13, ll. 16-25.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499⁴

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
2	2. The method of claim 1 wherein providing the input to the computer program comprises communicating information representative of the input to a graphical windowing environment.	“As shown [in Figure 3], the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.” Col. 8, ll. 8-13.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
3	3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.	<p>“The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract.</p> <p>“The alternative input subsystem module provides communication between the attached user friendly input devices and the remainder of the advanced user interface as well as application programs through the integrated operating environment. The alternative input subsystem module receives the input signals generated by the input devices and translates them to input messages useable in the advanced user interface.” Col. 3, ll. 57-65.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs.” Col. 7, ll. 15.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
4	4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same manner as if the input was received via a hardware keyboard.	<p>The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs.” Col. 7, ll. 15.</p> <p>“Another recent trend is to provide some sort of integration of computer program applications. Without integration, the user must employ separate application programs for word processing, database manipulation, graphics and electronic mail functions, and so forth. It is often quite difficult to integrate the outputs of the different programs into a single desired output. One solution has been to write a single integrated piece of software which incorporates a variety of applications which is called a multiple-function program.” Col. 2, ll. 35-44.</p> <p>“The interface profile module 104 is comprised of sets of application profiles 105 and the user profiles 107, which are files which list input messages produced by the AIS 103 from the input signals received by input devices 36, 38, 40, mapped to keyboard, mouse or other commands which are usable by existing application programs, e.g., mouse clicks, keystroke messages, MACROs, utility programs, etc.” Col. 7, ll. 66-68 to Col. 8, ll. 1-5.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving information corresponding to a keyboard character entered via the displayed keyboard.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 8, ll. 20-27.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
8	8. The method of claim 1 further comprising, hiding the input panel.	See Claims 1.1 – 1.6. <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include . . . an image magnifying utility . . . The image magnifying utility will magnify a rectangle of fixed size around a point at which the appropriate gesture was made. The utility allows very accurate positioning of a cursor in the expanded image. After stylus liftoff, the normal size display is restored, and the selected cursor coordinates are sent to the active application program.” Col. 8, ll. 14-36.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p> <p>In addition, the combination of Greanias ‘384 and U.S. Patent No. 5,148,155 (“Martin ‘155”) renders this claim obvious. Martin ‘155 teaches the hiding of an input panel. See Martin ‘155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
9	9. The method of claim 1 further comprising, docking the input panel.	<p>See Claim 25 of Greanias ‘384 (“The computer system as recited in claim 14, which further comprises a touch input device disposed over the viewing surface of the display device, and wherein the advanced user interface further comprises a pop-up keyboard module which displays an image of a keyboard on the display device[.]”) (emphasis added).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p> <p>In addition, the combination of Greanias ‘384 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified,</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
10	10. At least one computer-readable medium having computer-executable instructions, which when executed perform the method of claim 1.	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 5, ll. 52-63.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 5, ll. 52-63.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each executable input method is an interchangeable software component distinct from one or more application programs, each executable input method having a defined interface set such that the executable input method is connectable to the application programs;	<p>See Claims 1.1 – 1.6.</p> <p>The invention provides for multiple input types which supply user input to the computer system. See Figure 3.</p> <p>The input methods are each separately executable from the application programs. Col. 7, ll. 25 (“the AUI is written as an application program”).</p> <p>The input methods are connectable to the application programs through an advanced user interface and/or the operating system. Col. 1, ll. 5-11 (this invention “relates to an advanced user interface which allows a user to select one or more input devices to input data into a computer running a program originally written for a different input device in a multiapplication environment”). See also Figure 3.</p> <p>“The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract.</p>
11.2	opening an input window on a display of the computer system independent of a window of an active application program; and	<p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23.</p> <p>The AUI Utility is independent of the application program. Col. 8, ll. 8-13 (“the AUI utility can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment to communicate with other application programs”).</p>
11.3	displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input received by the selected executable input method via the interactive input panel is	<p>See Claims 1.1 – 1.6.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
	provided to the active application program as if the information was received via user input at a hardware input device.	gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 14-23 (emphasis added). The input methods are each separately executable from the application programs. Col. 7, ll. 25 (“the AUI is written as an application program”). “After receiving an input message from Presentation Manager.TM. 156 together with information about which is the active application program, the PM-Link 201 refers back to the alternate input subsystem 203 to determine whether the “keyboard” or “mouse” message it received is in fact a keyboard or a mouse message, or rather a touch, voice, image, or other message. The PM-Link 201 then passes the true message to the appropriate application program.” Col. 13, ll. 16-25.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
13	13. The computer-readable medium of claim 11 further comprising, providing a Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuable to display a selectable list of the plurality of executable input methods.	“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract. “It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 30-32. “A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 14-23 (emphasis added). The input methods are each separately executable from the application programs. Col.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		7, ll. 25 (“the AUI is written as an application program”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
14	14. The computer-readable medium of claim 13 further comprising, receiving a selection of one of the plurality of executable input methods displayed in the list as a selected executable input method, and in response, closing any open input window, and opening a new input window corresponding to the selected executable input method.	See Claim 8. “A few examples of utilities which might be found in the advanced user interface utilities 109 include . . . an image magnifying utility . . . The image magnifying utility will magnify a rectangle of fixed size around a point at which the appropriate gesture was made. The utility allows very accurate positioning of a cursor in the expanded image. After stylus liftoff, the normal size display is restored, and the selected cursor coordinates are sent to the active application program.” Col. 8, ll. 14-36.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
15	15. At least one computer-readable medium having computer-executable instructions, which when executed perform steps, comprising:	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	<p>means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 5, ll. 52-63.</p> <p>See Claims 1.1 – 1.6.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 14-23 (emphasis added).</p> <p>The AUI Utility is independent of the application program. Col. 8, ll. 8-13 (“the AUI utility can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment to communicate with other application programs”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement an actuatable icon representative of an input method list. For example, the invention discloses several types of interface utilities 109, and it would have been obvious to one of skill in the art to include this step to permit the user to select one of those input utilities. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p>
15.2	invoking a selected input method in response to a user selecting an icon corresponding to the selected input method, including presenting an input panel window; and	<p>See Claims 1.1 – 1.6.</p> <p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 8, ll. 20-27 (emphasis added).</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	See Claims 15.2.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
16	16. The computer-readable medium of claim 15 wherein accepting user data includes detecting user interaction with a touch-sensitive display.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 8, ll. 20-27 (emphasis added). See Claim 17 of Greanias ‘384 (“The computer system as recited in claim 14, which further comprises a gesture recognition unit which interprets the input signals transmitted by a touch input device in response to a user drawing symbols on a surface detected by the touch input device.”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
17	17. The computer-readable medium of claim 15 wherein each input method	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
	comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.	<p>objects. Col. 3, ll. 13-27; Col. 8, ll. 13-27. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Greanias ‘384 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p><u>The combination of Greanias ‘384 and U.S. Patent No. 5,946,499 (“the Saunders Reference”) also renders this claim obvious. In the Saunders Reference, text input methods communicate with the text services manager (“TSM”) through an object model interface. See Saunders Reference, Col. 4, ll. 23-43 (“Each second data structure TSMContext 40 is suitably attached to the first data structure TSMDocument 41 and contains two pieces of information, a LocaleObject 42 and a SOM (system object model) object 44. A LocaleObject 42 appropriately represents a text service and supplies the TSM 20 with the information necessary to load that text service,</u></p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u> including a text service class used by the TSM 20 to access the text service and a SOM class name for the SOM class implemented by the text service, as is well understood by those skilled in the art. A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
-------	-----------------------	---

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	“The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		<p>modules of the advanced user interface itself.” Abstract.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p> <p>In addition, the combination of Greanias ‘384 and U.S. Patent No. 5,454,046 (“the Carman Reference”)⁵ renders this claim obvious. See Carman Reference, Col. 10, ll. 14-22 (“The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation defined by the Unicode Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all languages of the world, whereas ASCII and ANSI representations are limited to 256 characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>See Figure 3A of Carman Reference.</p>

⁵ “Universal Symbolic Handwriting Recognition System” issued to Carman; filed September 17, 1993; issued September 26, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		<div data-bbox="289 478 717 987"> </div> <p data-bbox="755 693 792 819">FIG. 3A</p> <p data-bbox="833 216 971 1281">Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p data-bbox="995 186 1317 1281">Similarly, the combination of Greanias ‘384 and U.S. Patent No. 5,455,901 (“the Friend Reference”)⁶ renders this claim obvious. See Claim 6 of Friend Reference (“The device of claim 5 wherein said translator comprises a generator for generating machine readable characters from said strokes.”); Claim 7 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise Unicode characters.”); Claim 8 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise ASCII characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an</p>

⁶ “Input Device With Deferred Translation” issued to Friend; filed September 12, 1994; issued October 3, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 6,339,780	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
19	19. In a computing environment, a system comprising,	“An advanced user interface for use with a computer system operating on an integrated operating environment.” Abstract.
19.1	a manager component stored on one or more computer-readable media and configured: to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable software component distinct from one or more computer programs, and to send input data corresponding to a user input received at the selected input method to a graphical windowing	“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract. “It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 30-32. The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The input methods are each separately executable from the application programs. Col 7, ll. 25 (“the AUI is written as an application program”); Col. 8, ll. 8-13 (“the AUI utility can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment to communicate with other application programs”). An example input

Claim	U.S. Patent 6,339,780	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
	environment; and	method application is a “pop-up keyboard.” Col. 8, ll. 5-8 (“The Advanced User Interface utility (AUI utility) 109 is a set of utility programs such as a pop-up keyboard or an image magnifier utility which can be used with AUI 100.”). The AUI has “translation capabilities” for sending user data to the application programs. Col. 7, ll. 15.
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	The Integrated Operating Environment (windowing environment) serves as the interface between the AUI and the applications. See Figures 3 & 6; see also Abstract (“The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.”). The AUI Utility is independent of the application program. Col. 8, ll. 8-13 (“the AUI utility can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment to communicate with other application programs”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	“The integrated operating environment allows a plurality of application programs to be running simultaneously, one of which is designated the active application program to which all input data is directed.” Abstract.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
21	21. The system of claim 19 further comprising an input panel window corresponding to the selected input method.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
23	23. The system of claim 21 wherein the manager component selectively displays and hides the input panel window.	See Claim 8. “By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
-------	-----------------------	--

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
25	25. The system of claim 19 where the input method is displayed on a touch-sensitive display screen.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
26	26. The system of claim 19 wherein the manager component transfers information from the computer program to the selected input method.	See Figure 8. “After being recognized by the gesture recognition unit 213 as a circle gesture, the PM-Link 201 passes the circle gesture to the spreadsheet 150. The spreadsheet returns an “R0” message at 285, indicating that the circle gesture was not understood. Since the circle gesture was not understood at 287, the PM-Link 201 refers to the application profile 205 for the spreadsheet 150 at 289 and finds no corresponding command for the circle gesture. The PM-Link 201 then refers to the user profile 207 at 291 which contains the corresponding command “invoke the pop-up keyboard”. As the command from the user profile 207 has the highest priority at 293, the PM-Link 201 sends this message to the AUI utilities module 209 at 295 and the pop-up keyboard is presented to the user on the display.” Col. 16, ll. 17-31.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined	See Figure 4. See Claim 37 of Greanias ‘384 (“The set of interface profiles as recited in claim 31, wherein the interface profiles contain mappings of touch input messages against

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
	interface set.	corresponding mouse commands.”).
		“After determining the active application program, spreadsheet 50 at 119, the environment link 101 refers to the application profile 105 of the spreadsheet 50 for the command which corresponds to the input message “GRAPH” which will be recognized by the spreadsheet 50, i.e., the menu selection-- graph--.” Col. 8, ll. 59-65.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
28	28. The system of claim 19 wherein the selected input method comprises an object.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 3, ll. 13-27; Col. 8, ll. 13-27. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Greanias ‘384 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u> execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p><u>The combination of Greanias ‘384 and U.S. Patent No. 5,946,499 (“the Saunders Reference”) also renders this claim obvious. In the Saunders Reference, text input methods communicate with the text services manager (“TSM”) through an object model interface. See Saunders Reference, Col. 4, ll. 23-43 (“Each second data structure TSMContext 40 is suitably attached to the first data structure TSMDocument 41 and contains two pieces of information, a LocaleObject 42 and a SOM (system object model) object 44. A LocaleObject 42 appropriately represents a text service and supplies the TSM 20 with the information necessary to load that text service, including a text service class used by the TSM 20 to access the text service and a SOM class name for the SOM class implemented by the text service, as is well understood by those skilled in the art. A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</u></p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need</p>
-------	-----------------------	---

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
29	29. The system of claim 19 wherein the selected input method draws an input panel in an input panel window displayed in the graphical windowing environment.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
30	30. The system of claim 29 wherein the manager component selectively displays and hides the display of the input panel window.	See Claim 8. “By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
31	31. The system of claim 29 wherein the	See Claim 9.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
	manager component docks the input panel window.	<p>See Claim 25 of Greanias ‘384 (“The computer system as recited in claim 14, which further comprises a touch input device disposed over the viewing surface of the display device, and wherein the advanced user interface further comprises a pop-up keyboard module which displays an image of a keyboard on the display device[.]”) (emphasis added).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p> <p>In addition, the combination of Greanias ‘384 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

EXHIBIT I

Exemplar Chart of U.S. Patent 7,411,582**U.S. Patent 5,148,155 (“Martin ‘155”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499⁴
1	1. In a computing environment, a computer-implemented method comprising:	“A computer system having a digitizing tablet overlaying the display screen.” Abstract.
1.1	displaying an actuatable icon representative of an input method list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a	The DEVICES button is an actuatable icon. See Figure 1.

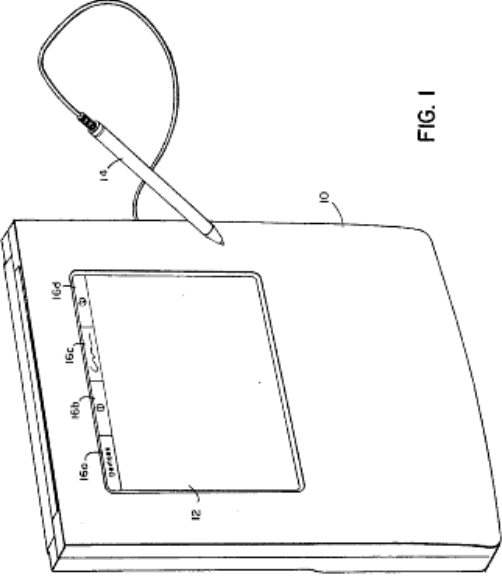
NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

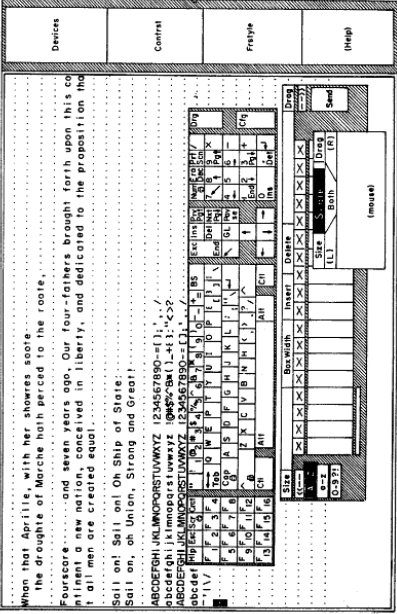
¹ “Computer With Tablet Input To Standard Programs” issued to Martin et al.; filed November 13, 1990; issued September 15, 1992.

² Kraig Brockschmidt, Inside OLE (2d ed. 1995).

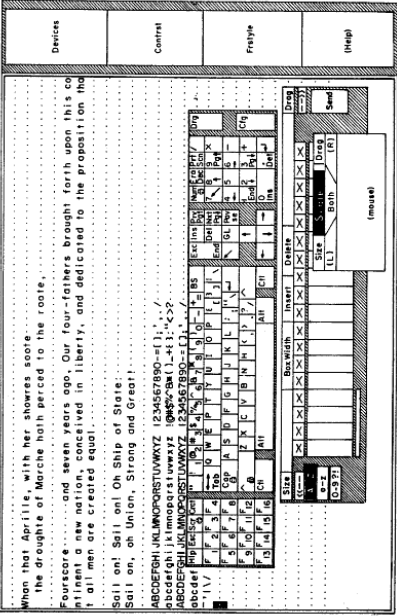
³ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

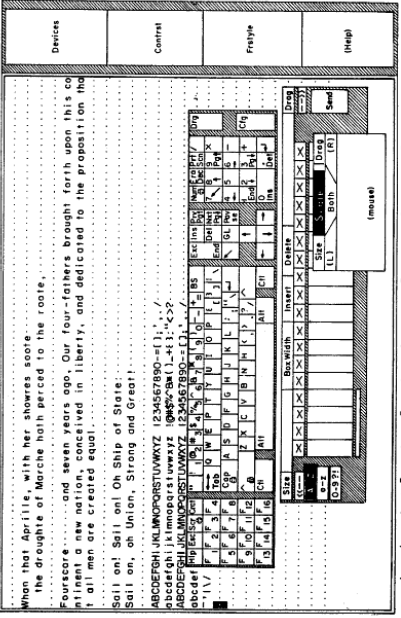
⁴ “Method And Apparatus For Processing Text Inputs From Multiple Input Devices In A Plurality Of Applications” issued to Saunders; filed May 10, 1996; issued August 31, 1999.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499 ⁴
	computer-executable software component distinct from the computer programs;	 <p style="text-align: center;">FIG. 1</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed.” Col. 17, ll. 11-15.</p> <p>“[T]hese devices are independent of the various application programs with which they may be used. In the preferred embodiment, they are even implemented in part using a processor separate from the processor on which the application is running.” Col. 16, ll. 30-35.</p>
1.2	in response to actuation of the actuatable icon, displaying the input method list;	<p>See Claim 1.1 regarding the actuation of the actuatable icon.</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed.” Col. 17, ll. 11-15.</p> <p>See Figure 10.</p>

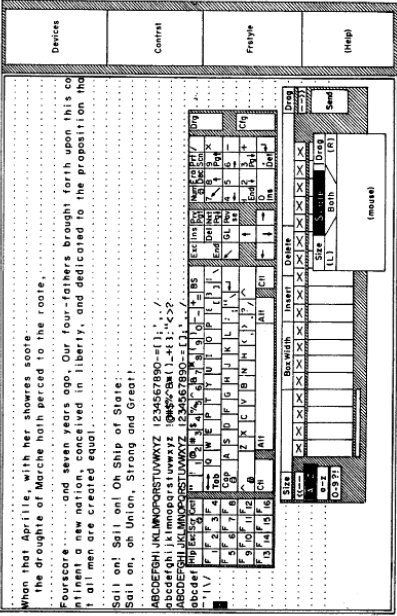
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499 ⁴
1.3	receiving a selection of an input method from the input method list;	 <p>FIG. 10</p>
		<p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active[.]” Col. 17, ll. 11-16.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499 ⁴
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	<p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.</p> <p>Once installed, the input method may communicate with the active application. Col. 18, ll. 12-16 (“FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data.”).</p>
1.5	receiving input via the interactive input panel; and	See Figure 10.

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt² and/or the MessagePad 120³ and/or U.S. Patent No. 5,946,499⁴</p>
		 <p>FIG. 10</p> <p>“The simulated keyboard device allows users to enter keyboard data by ‘typing’ with the stylus on an iconic keyboard.” Col. 27, ll. 54-56.</p> <p>“The primary body of the icon for the handwriting recognition input device contains boxes into which the user can write characters.” Col. 21, ll. 46-48.</p>
1.6	providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.	<p>“Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor’s BIOS 212 and services provided by the interface processor’s firmware 214.” Col. 34, ll. 8-15.</p> <p>“FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data.” Col. 18, ll. 12-16.</p> <p>See Figure 10.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499 ⁴
		 <p style="text-align: center;">FIG. 10</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
2	2. The method of claim 1 wherein providing the input to the computer program comprises communicating information representative of the input to a graphical windowing environment.	<p>“Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor’s BIOS 212 and services provided by the interface processor’s firmware 214.” Col. 34, ll. 8-15.</p> <p>“FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data.” Col. 18, ll. 12-16.</p> <p>See Figure 10.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
3	3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.	<p>"In order to effect this overlaying <i>interface</i> without interfering with the execution of the pre-existing program, an interface processor is provided to execute software used in implementing this interface." Col. 3, ll. 37-40 (emphasis added).</p> <p>"Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor's BIOS 212 and services provided by the interface processor's firmware 214." Col. 34, ll. 8-15.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
4	4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same manner as if the input was received via a hardware keyboard.	“Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor’s BIOS 212 and services provided by the interface processor’s firmware 214.” Col. 34, ll. 8-15. “The interface processor manages input from the tablet . . . and provides keystroke and mouse data to the main processor as if from a standard keyboard controller.” Abstract.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving information corresponding to a keyboard character entered via the displayed keyboard.	“The simulated keyboard device allows users to enter keyboard data by ‘typing’ with the stylus on an iconic keyboard.” Col. 27, ll. 54-56. See Figure 10.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		<p>FIG. 10</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
8	8. The method of claim 1 further comprising, hiding the input panel.	<p>“During interaction with one of the user’s programs, the user can activate and deactivate simulated devices (by removing them from and returning them to a device tray)[.]” Abstract.</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’. ” Col. 17, ll. 11-30.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
9	9. The method of claim 1 further comprising, docking the input panel.	“As with the other device icons, the keyboard icon includes a drag sub-icon, by which the user can position the keyboard icon on the display.” Col. 29, ll. 1-4.

See Figure 10.

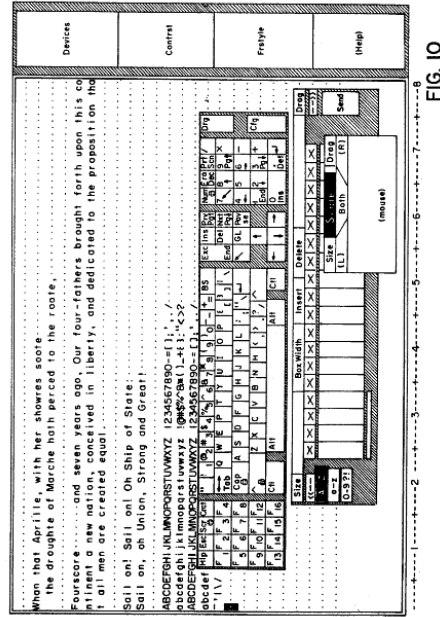


FIG. 10

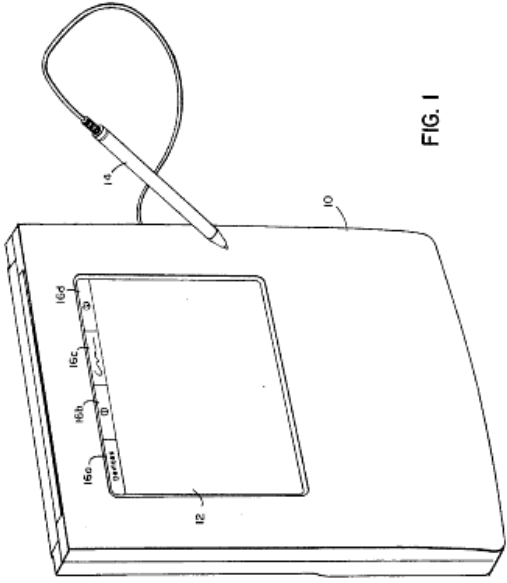
To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 55, ll. 55-68 to Col. 56, ll. 55-56.

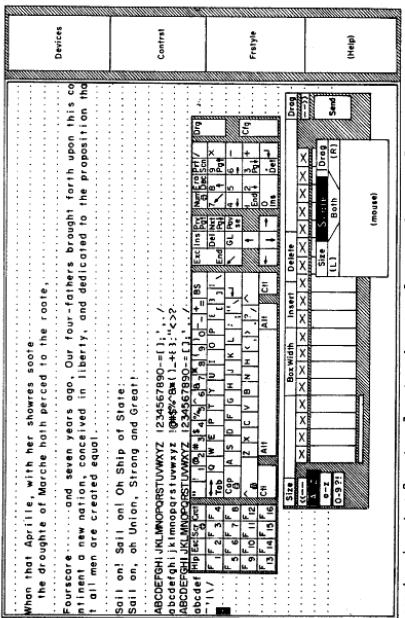
In addition, the combination of Martin ‘155 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field

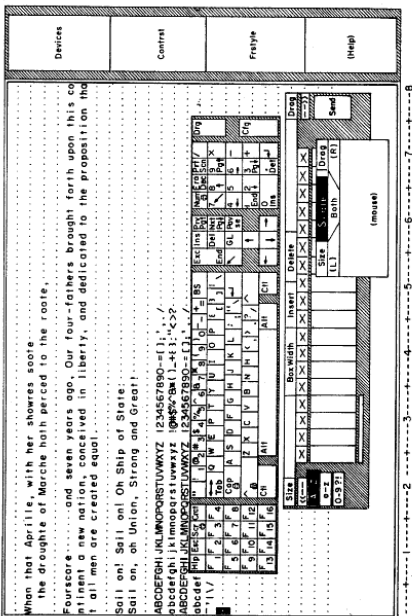
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

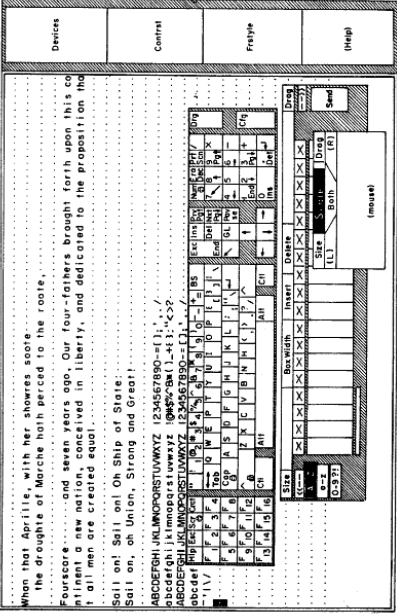
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
10	10. At least one computer-readable medium having computer-executable instructions, which when executed perform the method of claim 1.	“In order to effect this overlaying interface without interfering with the execution of the pre-existing program, an interface processor is provided to execute software used in implementing this interface.” Col. 3, ll. 37-40.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	“In order to effect this overlaying interface without interfering with the execution of the pre-existing program, an interface processor is provided to execute software used in implementing this interface.” Col. 3, ll. 37-40.

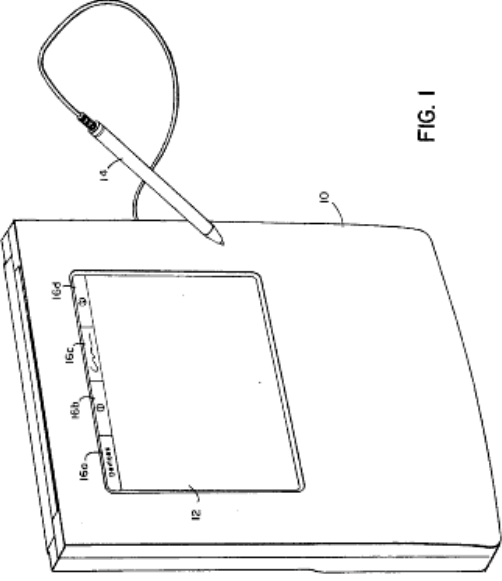
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each executable input method is an interchangeable software component distinct from one or more application programs, each executable input method having a defined interface set such that the executable input method is connectable to the application programs;	<p data-bbox="329 1115 362 1283">See Figure 1.</p>  <p data-bbox="805 527 829 579">FIG. 1</p> <p data-bbox="1000 1094 1032 1283">See Figure 10.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
11.2	opening an input window on a display of	 <p>FIG. 10</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.</p> <p>“[T]hese devices are independent of the various application programs with which they may be used. In the preferred embodiment, they are even implemented in part using a processor separate from the processor on which the application is running.” Col. 16, ll. 30-35.</p> <p>“FIG. 10 is a screen display showing the icons for three simulated devices. Behind</p>

<p>Claim</p>	<p>U.S. Patent 7,411,582</p>	<p>U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499</p>	<p>the computer system independent of a window of an active application program; and</p> <p>these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data." Col. 18, ll. 12-16.</p> <p>See Figure 10.</p>  <p>FIG. 10</p>	<p>11.3</p> <p>displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input received by the selected executable input method via the interactive input panel is provided to the active application program as if the information was received via user input at a hardware input device.</p> <p>"FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data." Col. 18, ll. 12-16.</p> <p>See Figure 10.</p>
--------------	------------------------------	--	--	---

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		 <p>FIG. 10</p>
		<p>“The interface processor manages input from the tablet . . . and provides keystroke and mouse data to the main processor as if from a standard keyboard controller.” Abstract.</p>

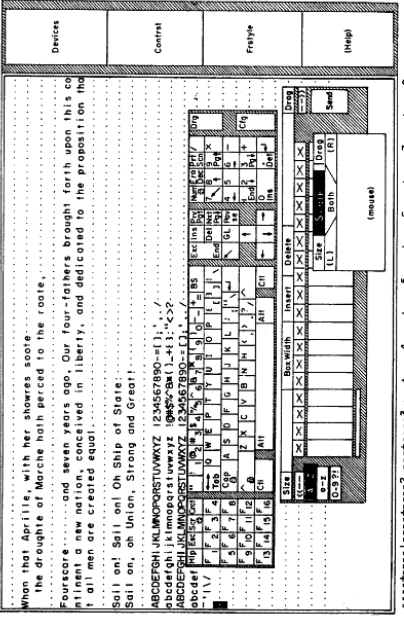
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
13	13. The computer-readable medium of claim 11 further comprising, providing a Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuable to display a selectable list of the plurality of executable input methods.	<p>The DEVICES button is the SIP menu button.</p> <p>See Figure 1.</p>

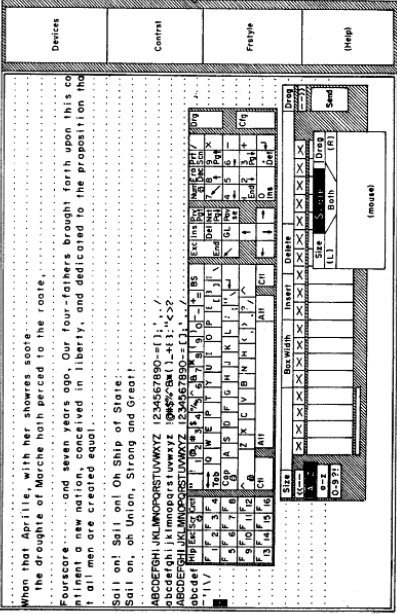
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		 <p style="text-align: center;">FIG. 1</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed.” Col. 17, ll. 11-15.</p>

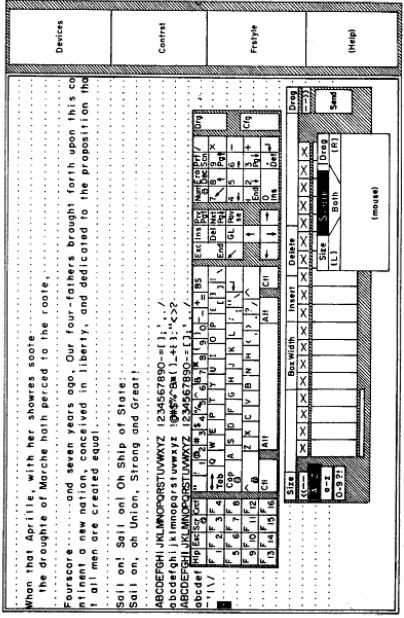
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
14	14. The computer-readable medium of claim 13 further comprising, receiving a selection of one of the plurality of executable input methods displayed in the list as a selected executable input	<p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
	method, and in response, closing any open input window, and opening a new input window corresponding to the selected executable input method.	results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
15	15. At least one computer-readable medium having computer-executable instructions, which when executed perform steps, comprising:	“In order to effect this overlaying interface without interfering with the execution of the pre-existing program, an interface processor is provided to execute software used in implementing this interface.” Col. 3, ll. 37-40.
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	See Figure 10.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
15.2	invoking a selected input method in response to a user selecting an icon	 <p>FIG. 10</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.</p> <p>“[T]hese devices are independent of the various application programs with which they may be used. In the preferred embodiment, they are even implemented in part using a processor separate from the processor on which the application is running.” Col. 16, ll. 30-35.</p> <p>See Figure 10.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
	corresponding to the selected input method, including presenting an input panel window; and	 <p>FIG. 10</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	<p>"FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data." Col. 18, ll. 12-16.</p> <p>See Figure 10.</p>  <p>FIG. 10</p> <p>"The interface processor manages input from the tablet . . . and provides keystroke and mouse data to the main processor as if from a standard keyboard controller." Abstract.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
16	16. The computer-readable medium of claim 15 wherein accepting user data includes detecting user interaction with a	<p>See Figure 1.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
	touch-sensitive display.	<div data-bbox="300 438 787 1005" data-label="Image"> </div> <div data-bbox="701 529 727 585" data-label="Caption">FIG. 1</div> <div data-bbox="855 1092 896 1285" data-label="Text">See Figure 10.</div> <div data-bbox="904 426 1318 1039" data-label="Image"> </div> <div data-bbox="1318 464 1344 529" data-label="Caption">FIG. 10</div> <div data-bbox="1385 289 1466 1285" data-label="Text"> <p>"A computer system having a digitizing tablet overlaying the display screen." Abstract.</p> </div>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
17	17. The computer-readable medium of claim 15 wherein each input method comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 55, ll. 55-68 to Col. 56, ll. 55-56. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Martin ‘155 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		<p>virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p><u>The combination of Martin ‘155 and U.S. Patent No. 5,946,499 (“the Saunders Reference”) also renders this claim obvious. In the Saunders Reference, text input methods communicate with the text services manager (“TSM”) through an object model interface. See Saunders Reference, Col. 4, ll. 23-43 (“Each second data structure TSMContext 40 is suitably attached to the first data structure TSMDocument 41 and contains two pieces of information, a LocaleObject 42 and a SOM (system object model) object 44. A LocaleObject 42 appropriately represents a text service and supplies the TSM 20 with the information necessary to load that text service, including a text service class used by the TSM 20 to access the text service and a SOM class name for the SOM class implemented by the text service, as is well understood by those skilled in the art. A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although described as a SOM object, i.e., a mechanism created in accordance with SOM, from using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</u></p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value. Col. 55, ll. 55-68 to Col. 56, ll. 55-56.</p> <p>In addition, the combination of Martin ‘155 and U.S. Patent No. 5,454,046 (“the Carman Reference”)⁵ renders this claim obvious. See Carman Reference, Col. 10, ll. 14-22 (“The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation defined by the Unicode Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all languages of the world, whereas ASCII and ANSI representations are limited to 256 characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>See Figure 3A of Carman Reference.</p>

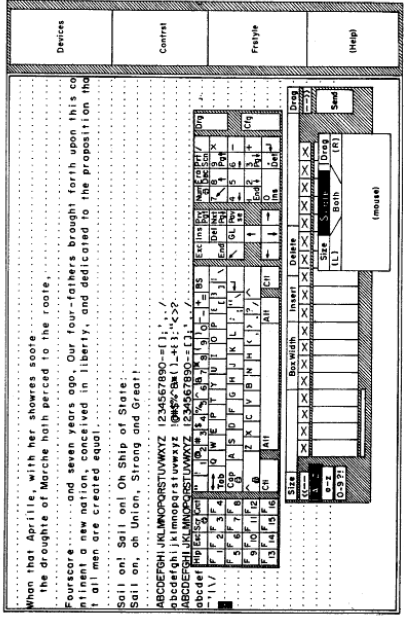
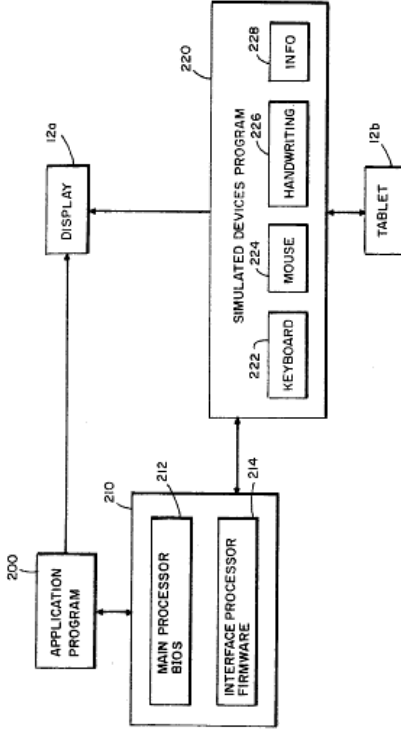
⁵ “Universal Symbolic Handwriting Recognition System” issued to Carman; filed September 17, 1993; issued September 26, 1995.

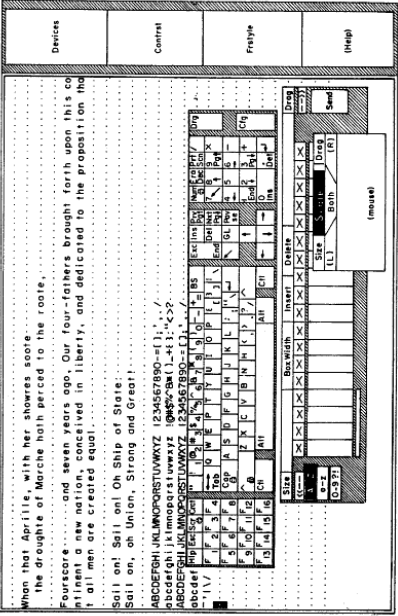
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		<div data-bbox="289 478 717 991"> </div> <p data-bbox="755 693 792 823">FIG. 3A</p> <p data-bbox="833 218 971 1281">Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p data-bbox="995 186 1315 1281">Similarly, the combination of Martin ‘155 and U.S. Patent No. 5,455,901 (“the Friend Reference”)⁶ renders this claim obvious. See Claim 6 of Friend Reference (“The device of claim 5 wherein said translator comprises a generator for generating machine readable characters from said strokes.”); Claim 7 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise Unicode characters.”); Claim 8 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise ASCII characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an</p>

⁶ “Input Device With Deferred Translation” issued to Friend; filed September 12, 1994; issued October 3, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 6,339,780	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
19	19. In a computing environment, a system comprising,	"A computer system having a digitizing tablet overlaying the display screen." Abstract.
19.1	a manager component stored on one or more computer-readable media and configured: to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable software component distinct from one or more computer programs, and to send input data corresponding to a user input received at the selected input method to a graphical windowing	See Figure 2. <p style="text-align: right;">FIG. 2</p>

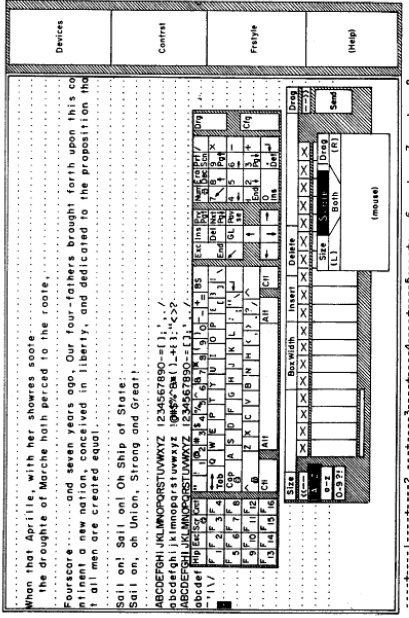
Claim	U.S. Patent 6,339,780	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
	environment; and	<p>See Figure 10.</p>  <p>FIG. 10</p> <p>See Figure 11.</p>  <p>FIG. 11</p> <p>“The interface processor <i>manages</i> input from the tablet . . . and provides keystroke and mouse data to the main processor as if from a standard keyboard controller.” Abstract (emphasis added).</p>

Claim	U.S. Patent 6,339,780	<p>U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u></p>
		<p>“Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor’s BIOS 212 and services provided by the interface processor’s firmware 214.” Col. 34, ll. 8-15.</p> <p>“FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data.” Col. 18, ll. 12-16.</p> <p>See Figure 10.</p> 

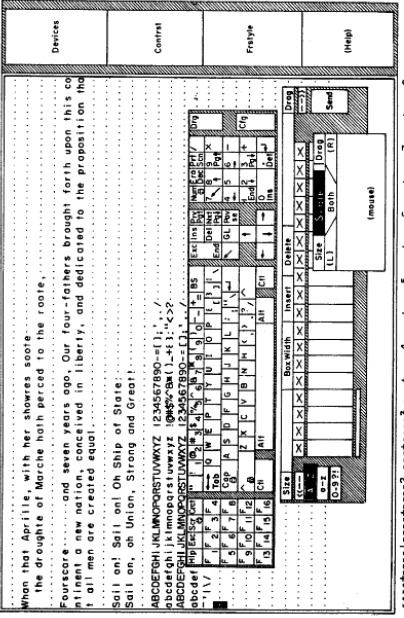
Claim	U.S. Patent 6,339,780	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	See Figures 2, 10 & 11. “The interface processor <i>manages</i> input from the tablet . . . and provides keystroke and mouse data to the main processor as if from a standard keyboard controller.” Abstract (emphasis added). “Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor’s BIOS 212 and services provided by the interface processor’s firmware 214.” Col. 34, ll. 8-15. “FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data.” Col. 18, ll. 12-16.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	See Figure 11.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		<div data-bbox="293 367 698 1092"> </div> <p data-bbox="714 357 738 430">FIG. 11</p> <p data-bbox="795 170 941 1289">“To receive data from the interface processor, an application must provide a routine that is to be called when data is received from the interface processor. The application registers this data handler routine with the datalink driver by the ‘install’ BIOS call mentioned above.” Col. 14, ll. 63-68.</p> <p data-bbox="974 170 1055 1289">“The BIOS for the main processor provides a mechanism by which an application running in the main processor can register to receive tablet data.” Col. 15, ll. 62-64.</p> <p data-bbox="1088 170 1201 1289">“Although block 200 is identified as “Application Program”, it represents any program (including system programs) with which a user may wish to interact via the tablet and display.” Col. 33, ll. 32-35.</p> <p data-bbox="1234 170 1347 1289">To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system to send user input to an application program having focus. Col. 55, ll. 55-68 to Col. 56, ll. 55-56.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
21	21. The system of claim 19 further comprising an input panel window corresponding to the selected input method.	See Figure 10. 
		“The simulated keyboard device allows users to enter keyboard data by “typing” with the stylus on an iconic keyboard.” Col. 27, ll. 54-56.

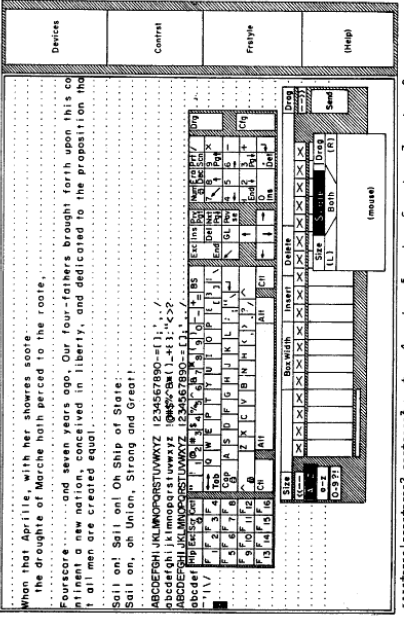
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	See Figure 10.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		 <p>FIG. 10</p>
		“The simulated keyboard device allows users to enter keyboard data by “typing” with the stylus on an iconic keyboard.” Col. 27, ll. 54-56.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
23	23. The system of claim 21 wherein the manager component selectively displays and hides the input panel window.	<p>“During interaction with one of the user’s programs, the user can activate and deactivate simulated devices (by removing them from and returning them to a device tray)[.]” Abstract.</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30. “Provisions can be made for an application to invoke user services provided by the user interface subsystem. An application running in the main processor can make a particular call, which causes the main processor to pass a code to the interface processor, identifying the service to be performed. For example, it is desirable for certain applications to be able to force certain simulated devices to appear and disappear at appropriate times.” Col. 32, ll. 36-44.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
25	25. The system of claim 19 where the input method is displayed on a touch-sensitive display screen.	See Figure 10.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		 <p>FIG. 10</p>
		<p>“The simulated keyboard device allows users to enter keyboard data by “typing” with the stylus on an iconic keyboard.” Col. 27, ll. 54-56.</p> <p>“A computer system having a digitizing tablet overlaying the display screen.” Abstract.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
26	26. The system of claim 19 wherein the manager component transfers information from the computer program to the selected input method.	<p>“Provisions can be made for an application to invoke user services provided by the user interface subsystem. An application running in the main processor can make a particular call, which causes the main processor to pass a code to the interface processor, identifying the service to be performed. For example, it is desirable for certain applications to be able to force certain simulated devices to appear and disappear at appropriate times.” Col. 32, ll. 36-44.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>

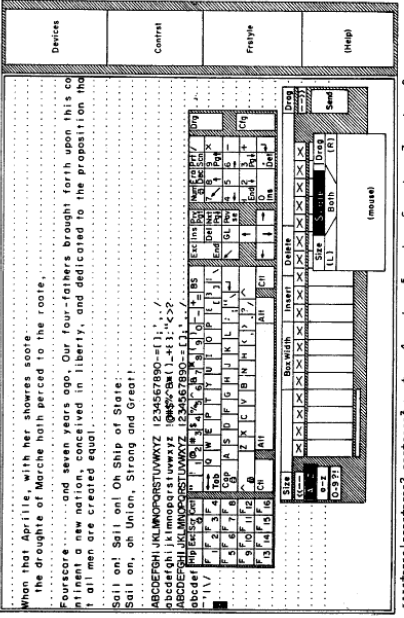
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined interface set.	<p>“In order to effect this overlaying <i>interface</i> without interfering with the execution of the pre-existing program, an interface processor is provided to execute software used in implementing this interface.” Col. 3, ll. 37-40 (emphasis added).</p> <p>“Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor’s BIOS 212 and services provided by the interface processor’s firmware 214.” Col. 34, ll. 8-15.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
28	28. The system of claim 19 wherein the selected input method comprises an object.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 55, ll. 55-68 to Col. 56, ll. 55-56. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u></p>
		<p>rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Martin ‘155 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p><u>The combination of Martin ‘155 and U.S. Patent No. 5,946,499 (“the Saunders Reference”) also renders this claim obvious. In the Saunders Reference, text input methods communicate with the text services manager (“TSM”) through an object model interface. See Saunders Reference, Col. 4, ll. 23-43 (“Each second data structure TSMContext 40 is suitably attached to the first data structure TSMDocument 41 and contains two pieces of information, a LocaleObject 42 and a SOM (system object model) object 44. A LocaleObject 42 appropriately represents a text service and supplies the TSM 20 with the information necessary to load that text service, including a text service class used by the TSM 20 to access the text service and a SOM class name for the SOM class implemented by the text service, as is well understood by those skilled in the art. A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although</u></p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		<u>described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), <u>disclose the same or similar techniques, and were developed during the same general time period.</u></u>
		Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
29	29. The system of claim 19 wherein the selected input method draws an input panel in an input panel window displayed in the graphical windowing environment.	See Figure 10.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		 <p>FIG. 10</p>
		<p>“The simulated keyboard device allows users to enter keyboard data by “typing” with the stylus on an iconic keyboard.” Col. 27, ll. 54-56.</p> <p>“The primary body of the icon for the handwriting recognition input device contains boxes into which the user can write characters.” Col. 21, ll. 46-48.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
30	30. The system of claim 29 wherein the manager component selectively displays and hides the display of the input panel window.	<p>“During interaction with one of the user’s programs, the user can activate and deactivate simulated devices (by removing them from and returning them to a device tray).” Abstract.</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user 'operates'." Col. 17, ll. 11-30.

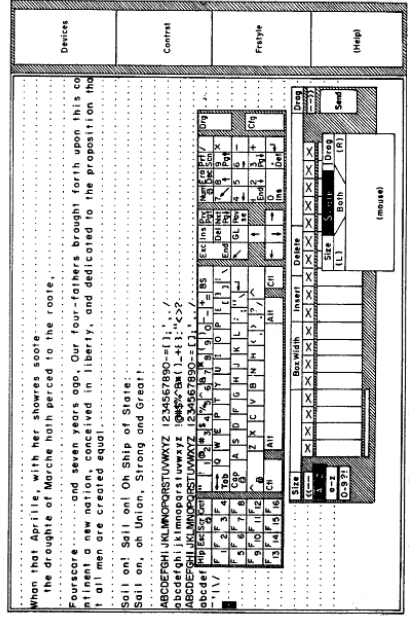
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
31	31. The system of claim 29 wherein the manager component docks the input panel window.	See Figure 10. 

FIG. 10

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		<p>“As with the other device icons, the keyboard icon includes a drag sub-icon, by which the user can position the keyboard icon on the display.” Col. 29, ll. 1-4.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 55, ll. 55-68 to Col. 56, ll. 55-56.</p> <p>In addition, the combination of Martin ‘155 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

EXHIBIT J

Exemplar Chart of U.S. Patent 7,411,582**Apple Newton MessagePad 120 (“MessagePad 120”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide ² and/or The NewtonScript Programming Language ³ and/or Brockschmidt ⁴ and/or U.S. Patent No. 5,946,499 ⁵
1	1. In a computing environment, a computer-implemented method comprising:	The MessagePad 120 is a handheld computer device running Newton OS that permits the user to select from among multiple input methods while entering data into an active application. These input methods are designed to work with a plurality of applications. The use of the various input methods of the MessagePad 120 with the Notepad and Calls

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

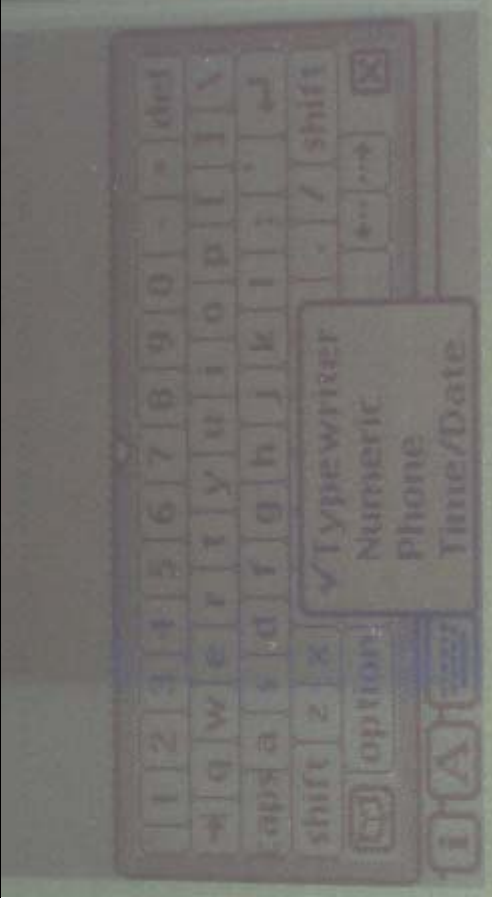
² Apple Press, Apple Computer Inc., Newton’s Programmer’s Guide, For Newton 2.0, Addison Wesley Publishing Company (1996).


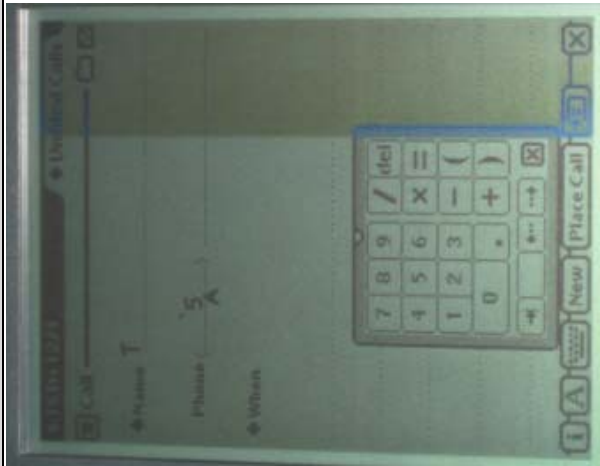
³ Apple Press, Apple Computer Inc., The NewtonScript Programming Language, Addison Wesley Publishing Company (1996).

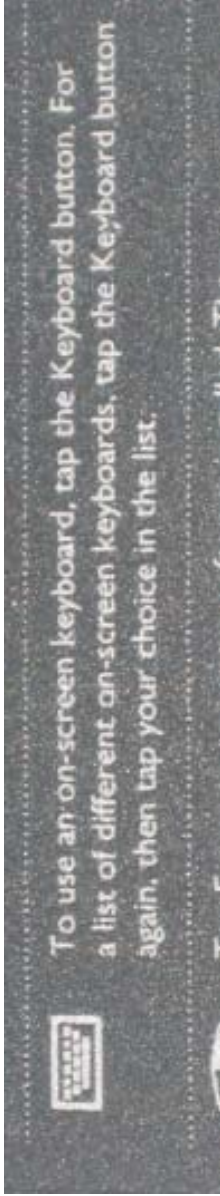
⁴ Kraig Brockschmidt, Inside OLE (2d ed. 1995).

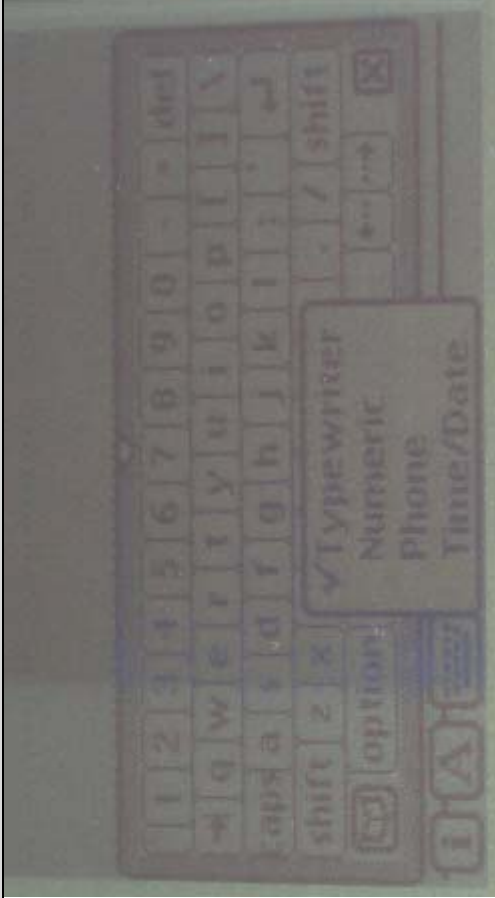

⁵ “Method And Apparatus For Processing Text Inputs From Multiple Input Devices In A Plurality Of Applications” issued to Saunders; filed May 10, 1996; issued August 31, 1999.

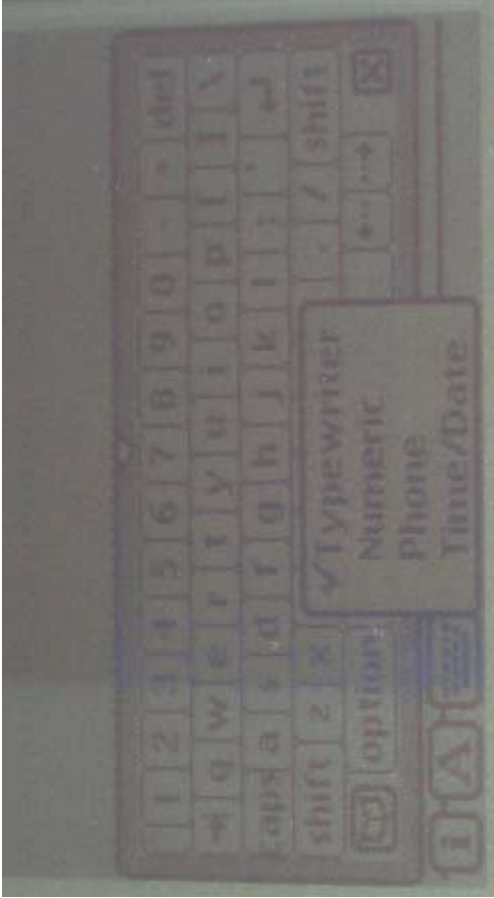
Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴ and/or U.S. Patent No. 5,946,499⁵ applications are shown in this chart.</p>
1.1	<p>displaying an actuatable icon representative of an input method list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a computer-executable software component distinct from the computer programs;</p>	<p>The screen cover of the MessagePad 120 describes the keyboard icon as follows:</p>  <p>The MessagePad 120 displays a keyboard icon along the status bar. As shown below, when pressed once, an on-screen keyboard appears; when pressed twice, the other available input methods are shown.</p> 


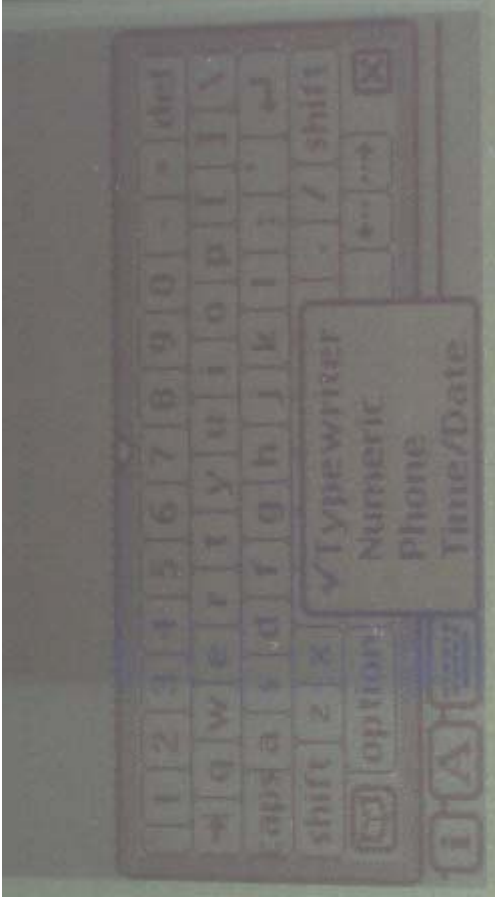
Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴ and/or U.S. Patent No. 5,946,499⁵</p>
		 <p>The input methods are distinct from the applications. As shown below, the on-screen Numeric keyboard works with the Notepad and Calls applications.</p>

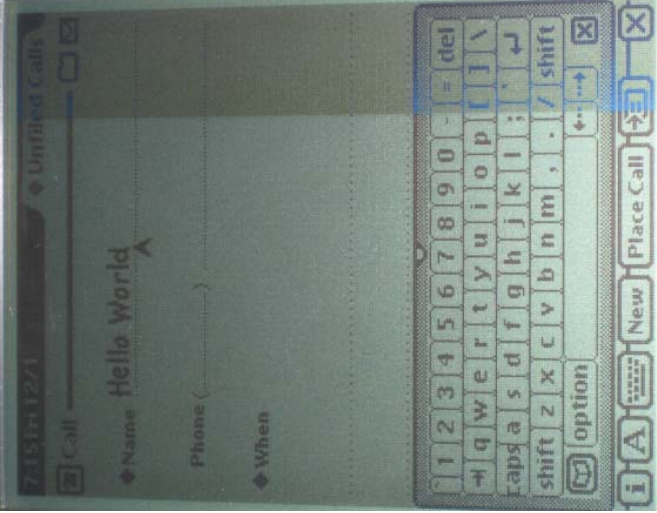
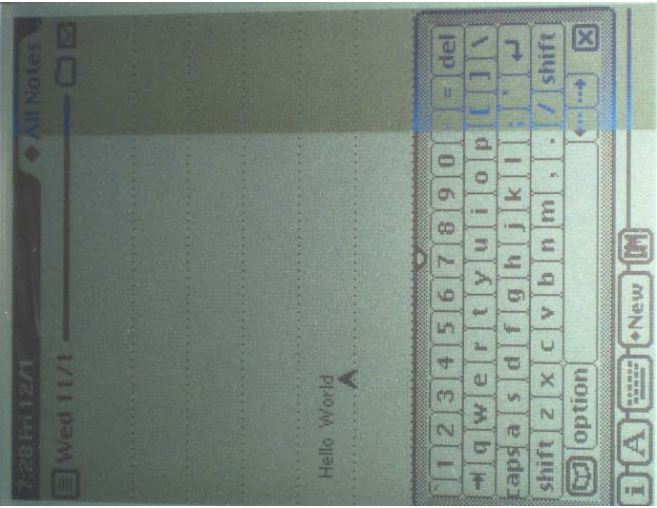
Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴ and/or U.S. Patent No. 5,946,499⁵</p>
		<div data-bbox="321 793 917 1270">  </div> <div data-bbox="321 304 917 766">  </div> <p>The Newton Programmer’s Guide further illustrates this capability. See Newton Programmer’s Guide at page 8-26:</p> <div data-bbox="1052 963 1099 1236"> <h3>Using Keyboards</h3> </div> <div data-bbox="1109 350 1206 1236"> <p>You can provide the user with on-screen keyboard input in your applications using the built-in keyboard views. You can also define new keyboard views and register them with the system, which will activate caret input when these views are opened.</p> </div> <div data-bbox="1229 1012 1269 1236"> <h3>Keyboard Views</h3> </div> <div data-bbox="1281 350 1347 1236"> <p>There are four different floating keyboards built into the system root view. Each of the built-in keyboards can be accessed as a child of the root with a symbol.</p> </div> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design an input system capable of working with virtually any application.</p>

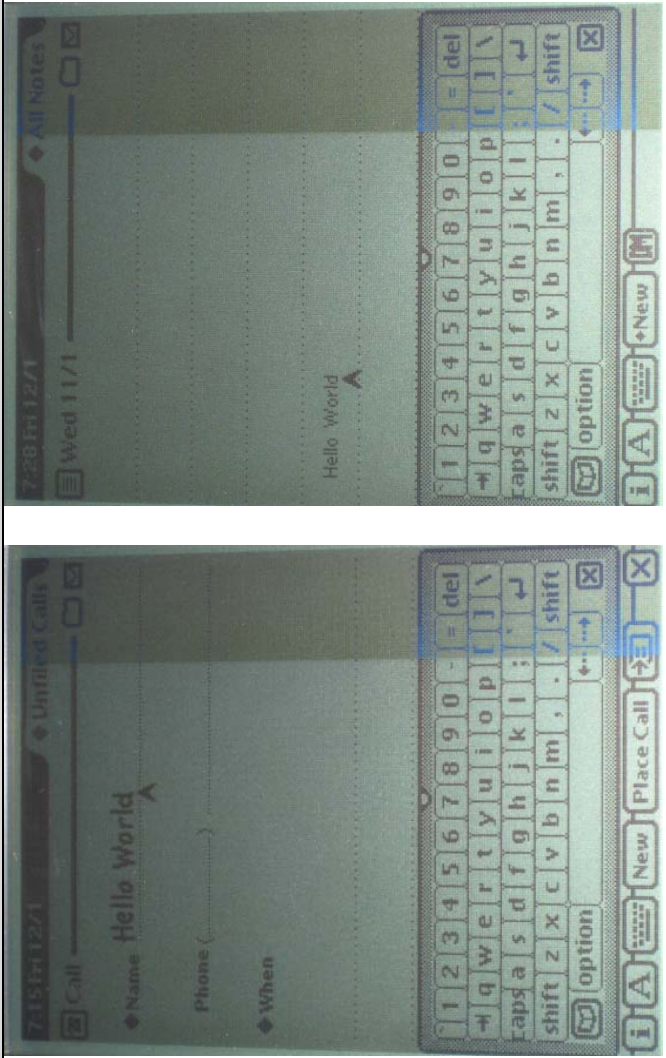
Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴ and/or U.S. Patent No. 5,946,499⁵</p> <p>In addition, the combination of the MessagePad 120 and U.S. Patent No. 5,148,155 (Martin ‘155) renders this claim obvious. Martin ‘155 teaches input methods which are independent of any application program. See Martin ‘155 Col. 16, ll. 30-32. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.2	in response to actuation of the actuatable icon, displaying the input method list;	<p>The screen cover of the MessagePad 120 describes the keyboard icon as follows:</p>  <p>As shown below, when the keyboard icon is pressed twice, the input method list appears:</p>

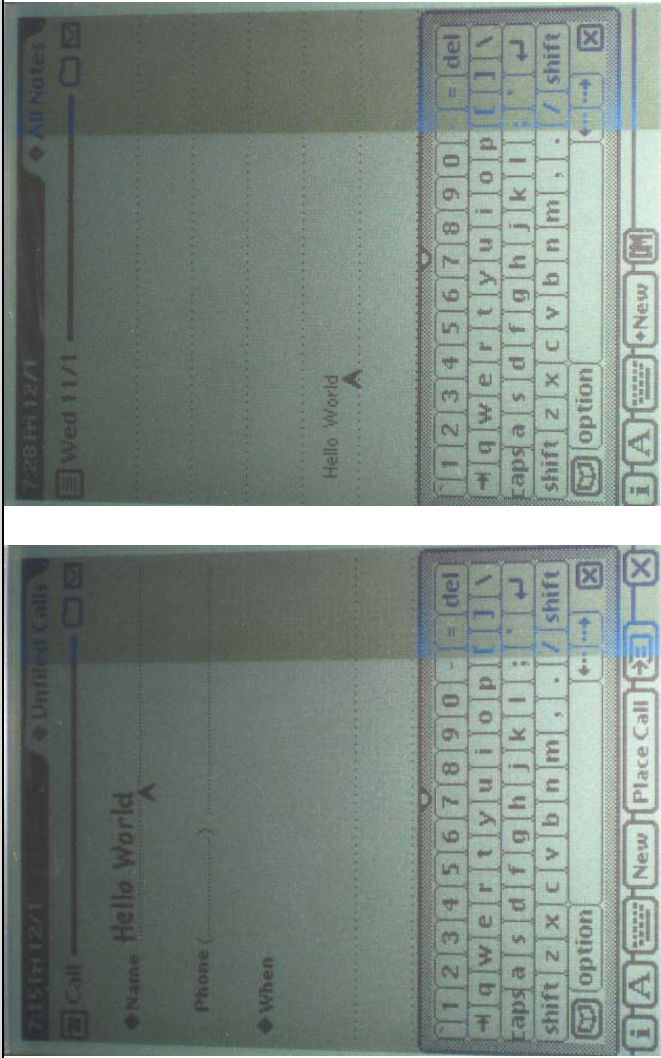
Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴ and/or U.S. Patent No. 5,946,499⁵</p> 
1.3	receiving a selection of an input method from the input method list;	<p>The MessagePad 120 displays a keyboard icon along the status bar. As shown below, when pressed once, an on-screen keyboard appears.</p> 

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴ and/or U.S. Patent No. 5,946,499⁵</p> <p>When pressed twice, the other available input methods are shown and can be selected.</p> 
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	<p>The MessagePad 120 displays a keyboard icon along the status bar. As shown below, when pressed once, an on-screen keyboard is installed.</p>

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴ and/or U.S. Patent No. 5,946,499⁵</p>
		<div data-bbox="321 590 813 984">  </div> <p>As shown below, once the keyboard input method is installed, the input method list shows that the keyboard input method has been installed.</p> <div data-bbox="971 342 1463 1234">  </div>

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴ and/or U.S. Patent No. 5,946,499⁵</p>
		<p>The keyboard input method, once installed, sends keystrokes to the active application. As shown below, the phrase “Hello World” appears in the Notes and Calls application windows when typed from the on-screen keyboard.</p> <div data-bbox="451 804 1104 1312">  </div> <div data-bbox="451 262 1104 766">  </div>
1.5	receiving input via the interactive input panel; and	As shown below, the phrase “Hello World” appears in the Notes and Calls application windows when typed from the on-screen keyboard.

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴ and/or U.S. Patent No. 5,946,499⁵</p> 
1.6	providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.	<p>Using the soft keyboard of the MessagePad 120, a user can send keystrokes to the active application. As shown below, the phrase “Hello World” appears in the Notes and Calls application windows when typed from the on-screen keyboard. The phrase “Hello World” appears on the application window as if the user had typed this phrase using a traditional keyboard.</p>

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴ and/or U.S. Patent No. 5,946,499⁵</p>
		

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499</p>
2	2. The method of claim 1 wherein providing the input to the computer program comprises communicating	See Claims 1.5 & 1.6.

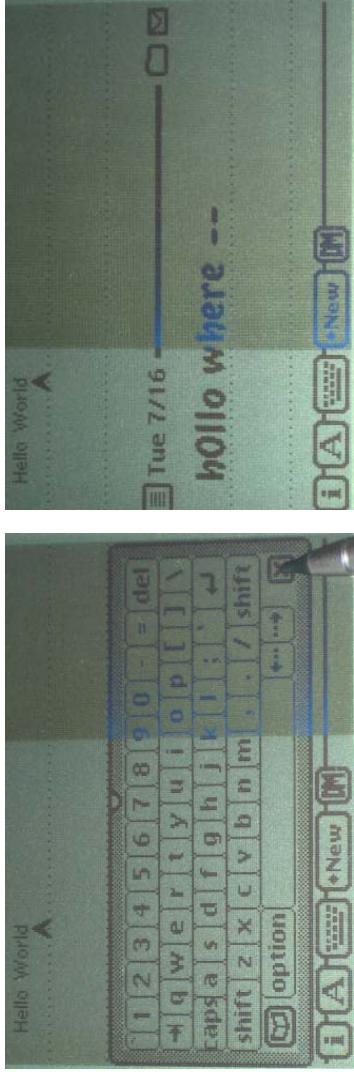
Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
	information representative of the input to a graphical windowing environment.	
Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
3	3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.	<p>See Claim 1.5.</p> <p>Because the on-screen keyboard is separate from the Notepad and Calls applications, the input data must pass through an interface before reaching the application.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design an input interface system between an input method and an application to permit any input method to work with virtually any application.</p> <p>In addition, the combination of the MessagePad 120 and U.S. Patent No. 5,148,155 ("Martin '155") renders this claim obvious. Martin '155 discloses a system with an interface between a plurality of input methods and the active application program. See Martin '155 Col 3, ll. 37-40. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need</p>


Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
		and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
4	4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same manner as if the input was received via a hardware keyboard.	See Claims 1 & 1.5.

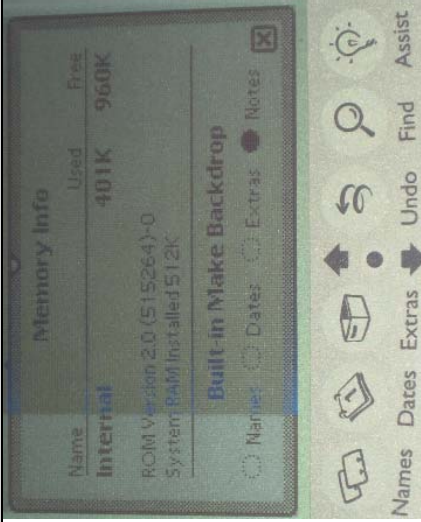

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
-------	-----------------------	---

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving information corresponding to a keyboard character entered via the displayed keyboard.	See Claims 1.5 & 1.6.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
8	8. The method of claim 1 further comprising, hiding the input panel.	When the close button of the input method window is depressed, the input method window closes. 

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
9	9. The method of claim 1 further comprising, docking the input panel.	See Claim 1.1, which illustrates the docking of the input panel as follows: 
		To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel.

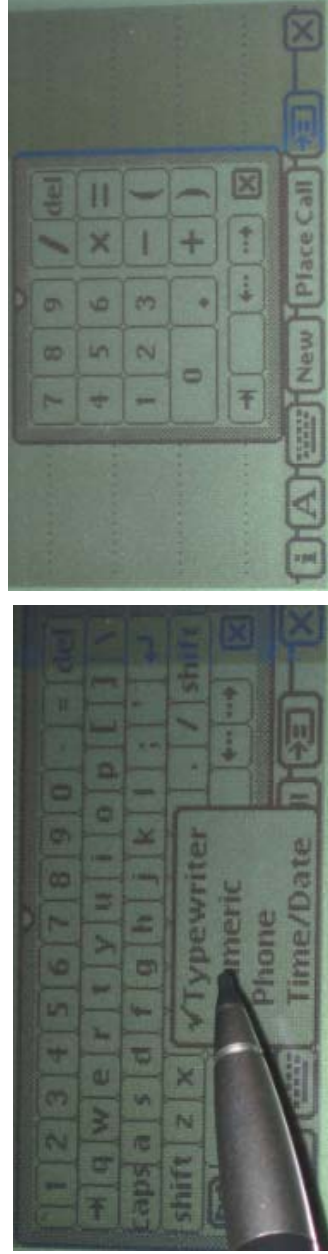
Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
10	10. At least one computer-readable medium having computer-executable instructions, which when executed	The MessagePad 120 has a computer-readable medium.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
	perform the method of claim 1.	 

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	See Claims 1.1 & 10.
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each executable input method is an	See Claim 1.1.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
	interchangeable software component distinct from one or more application programs, each executable input method having a defined interface set such that the executable input method is connectable to the application programs;	
11.2	opening an input window on a display of the computer system independent of a window of an active application program; and	See Claim 1.1.
11.3	displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input received by the selected executable input method via the interactive input panel is provided to the active application program as if the information was received via user input at a hardware input device.	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
13	13. The computer-readable medium of claim 11 further comprising, providing a Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuable to display a selectable list of the plurality of executable input methods.	See Claim 1.1.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
14	14. The computer-readable medium of claim 13 further comprising, receiving a selection of one of the plurality of executable input methods displayed in the list as a selected executable input method, and in response, closing any open input window, and opening a new input window corresponding to the selected executable input method.	As shown below, when the user selects the Numeric input method option, the Typewriter input method window closes, and the Numeric input method window opens. 

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
15	15. At least one computer-readable medium having computer-executable instructions, which when executed perform steps, comprising:	See Claims 1.1 & 10.
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	See Claim 1.1
15.2	invoking a selected input method in response to a user selecting an icon corresponding to the selected input method, including presenting an input panel window; and	See Claim 1.1.
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	See Claim 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
16	16. The computer-readable medium of claim 15 wherein accepting user data includes detecting user interaction with a touch-sensitive display.	See Claim 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
17	17. The computer-readable medium of claim 15 wherein each input method comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.	<p>NewtonScript, the programming language used to write software for the MessagePad 120, is built on an object model. See The NewtonScript Programming Language⁶, page 1-2:</p> <p>The Object Model</p> <p>NewtonScript is built on an object model. All data is stored as objects, or typed pieces of data. This differs from other object-oriented languages like C++ or Object Pascal, where data is a hybrid of objects and regular data types.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM</p>

⁶ Apple Press, Apple Computer Inc., The NewtonScript Programming Language, Addison Wesley Publishing Company (1996).

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499</p>
		<p>objects. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of the MessagePad 120 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>The combination of the MessagePad 120 and U.S. Patent No. 5,946,499 (“the Saunders Reference”) also renders this claim obvious. In the Saunders Reference, text input methods communicate with the text services manager (“TSM”) through an object model interface. See Saunders Reference, Col. 4, ll. 23-43 (“Each second data structure TSMContext 40 is suitably attached to the first data structure TSMDocument 41 and contains two pieces of information, a LocaleObject 42 and a SOM (system</p>

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt <u>and/or U.S. Patent No. 5,946,499</u>.</p> <p><u>object model) object 44. A LocaleObject 42 appropriately represents a text service and supplies the TSM 20 with the information necessary to load that text service, including a text service class used by the TSM 20 to access the text service and a SOM class name for the SOM class implemented by the text service, as is well understood by those skilled in the art. A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</u></p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	<p>The Newton programming environment supports Unicode. See Newton Programmer’s Guide at 1-9:</p> <p><i>The Newton text imaging facility supports Unicode directly, so the system can be easily localized to display languages using different script systems. The system is extensible, so it’s possible to add additional fonts, font engines, and printer drivers.</i></p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value.</p> <p>In addition, the combination of the MessagePad 120 and U.S. Patent No. 5,454,046 (“the Carmen Reference”)⁷ renders this claim obvious. See Carmen Reference, Col. 10, ll. 14-22 (“The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation defined by the Unicode Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all languages of the world, whereas ASCII and ANSI representations are limited to 256 characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>See Figure 3A of Carmen Reference.</p>

⁷ “Universal Symbolic Handwriting Recognition System” issued to Carman; filed September 17, 1993; issued September 26, 1995.

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499</p>
		<div data-bbox="365 472 795 997"> </div> <p style="text-align: center;"><u>FIG. 3A</u></p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>Similarly, the combination of the MessagePad 120 and U.S. Patent No. 5,455,901 (“the Friend Reference”)⁸ renders this claim obvious. See Claim 6 of Friend Reference (“The device of claim 5 wherein said translator comprises a generator for generating machine readable characters from said strokes.”); Claim 7 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise Unicode characters.”); Claim 8 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise ASCII characters.”). One of skill in the art</p>

⁸ “Input Device With Deferred Translation” issued to Friend; filed September 12, 1994; issued October 3, 1995.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
		would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.



Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
19	19. In a computing environment, a system comprising,	See Claim 1.
19.1	a manager component stored on one or more computer-readable media and configured: to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
	software component distinct from one or more computer programs, and to send input data corresponding to a user input received at the selected input method to a graphical windowing environment; and	
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	See Claims 1.1 & 1.5.
Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
21	21. The system of claim 19 further comprising an input panel window corresponding to the selected input method.	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
23	23. The system of claim 21 wherein the manager component selectively displays and hides the input panel window.	See Claim 14. See also page 24 of Newton Apple MessagePad Handbook, which provides:

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
		To use a different keyboard, tap the Keyboard button  a second time and from the list that appears, tap your choice. When you are finished using the on-screen keyboard, tap  on the bottom right to close it.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
25	25. The system of claim 19 where the input method is displayed on a touch-sensitive display screen.	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
26	26. The system of claim 19 wherein the manager component transfers information from the computer program to the selected input method.	See Claims 1.1 & 1.5. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design an input manager capable of transferring information from

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499</p> <p>the computer program to the selected input method.</p> <p>In addition, the combination of MessagePad 120 and U.S. Patent No. 5,148,155 (Martin ‘155) renders this claim obvious. Martin ‘155 teaches the transferring information from the computer program to the selected input method. See Martin ‘155 Col. 32, ll. 36-44. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
-------	-----------------------	--

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499</p>
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined interface set.	<p>See Claims 1.1 & 1.5.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system capable of calling functions in the manager component via a defined interface set.</p> <p>In addition, the combination of MessagePad 120 and U.S. Patent 5,157,384 (“Greanias</p>

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
		‘384”) renders this claim obvious. Greanias ‘384 teaches calling functions in the manager component via a defined interface set. See Greanias ‘384 Claim 37 and Col. 8, ll. 59-65. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
28	28. The system of claim 19 wherein the selected input method comprises an object.	NewtonScript, the programming language used to write software for the MessagePad 120, is built on an object model. See The NewtonScript Programming Language, page 1-2:

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499</p>
		<p>The Object Model</p> <p>NewtonScript is built on an object model. All data is stored as objects, or typed pieces of data. This differs from other object-oriented languages like C++ or Object Pascal, where data is a hybrid of objects and regular data types.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of the MessagePad 120 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with</p>

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499</p>
		<p>virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>The combination of the MessagePad 120 and U.S. Patent No. 5,946,499 (“the Saunders Reference”) also renders this claim obvious. In the Saunders Reference, text input methods communicate with the text services manager (“TSM”) through an object model interface. See Saunders Reference, Col. 4, ll. 23-43 (“Each second data structure TSMContext 40 is suitably attached to the first data structure TSMDocument 41 and contains two pieces of information, a LocaleObject 42 and a SOM (system object model) object 44. A LocaleObject 42 appropriately represents a text service and supplies the TSM 20 with the information necessary to load that text service, including a text service class used by the TSM 20 to access the text service and a SOM class name for the SOM class implemented by the text service, as is well understood by those skilled in the art. A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
29	29. The system of claim 19 wherein the selected input method draws an input panel in an input panel window displayed in the graphical windowing environment.	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499
30	30. The system of claim 29 wherein the manager component selectively displays and hides the display of the input panel window.	See Claim 23.


Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt and/or U.S. Patent No. 5,946,499</p>
31	<p>31. The system of claim 29 wherein the manager component docks the input panel window.</p>	<p>See Claim 1.1, which illustrates the docking of the input panel as follows:</p>  <p>See also page 55 of Newton Apple MessagePad Handbook, which provides:</p> <p>To move a keyboard, tap and hold down the pen on the picture hanger on the keyboard slip and drag to it where you want the keyboard on the screen.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel.</p>

EXHIBIT K

Exemplar Chart of U.S. Patent 7,411,582**U.S. Patent 5,252,951 (“Tannenbaum ‘951”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499 ⁴
1	1. In a computing environment, a computer-implemented method comprising:	“An advanced user interface for use with a computer system operating on an integrated operating environment.” Abstract.
1.1	displaying an actuatable icon representative of an input method	“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Graphical User Interface With Gesture Recognition In A Multiapplication Environment” issued to Tannenbaum et al.; filed October 21, 1991; issued October 12, 1993.

² Kraig Brockschmidt, Inside OLE (2d ed. 1995).

³ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

⁴ “Method And Apparatus For Processing Text Inputs From Multiple Input Devices In A Plurality Of Applications” issued to Saunders; filed May 10, 1996; issued August 31, 1999.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499⁴
	list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a computer-executable software component distinct from the computer programs;	<p>change to the application program code.” Abstract.</p> <p>“It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 38-40.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 3-12 (emphasis added).</p> <p>The input methods are each separately executable from the application programs. Col. 8, ll. 65-67 (“As shown, the AUI utility 109 can be written as a separate application program[.]”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement an actuable icon representative of an input method list. For example, the invention discloses several types of interface utilities 109, and it would have been obvious to one of skill in the art to include this step to permit the user to select one of those input utilities. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the presentation of an actuable icon representative of an input method list to a user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³ <u>and/or U.S. Patent No. 5,946,499⁴</u>
		combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.2	in response to actuation of the actuatable icon, displaying the input method list;	<p>See Claim 1.1 regarding the actuation of the actuatable icon.</p> <p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract.</p> <p>“It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 38-40.</p> <p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to present an input list to a user. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the presentation of an input list to a user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.3	receiving a selection of an input method from the input method list;	See Claim 1.2.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499 ⁴
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	<p>A user can install an input method by using the appropriate method. Col. 9, ll. 9-12. (“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.”).</p> <p>Once installed, the input method can use the translation capabilities of the AUI 100 to communicate with the active application. Col. 8, ll. 3. See also Col. 9, ll. 12-19 (“A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard. The space needed for the keyboard or keypad is quite small, and will not completely obscure the work area of the spreadsheet, word processor, etc., which is the active application program.”).</p>
1.5	receiving input via the interactive input panel; and	<p>Once installed, the input method can use the translation capabilities of the AUI 100 to communicate with the active application. Col. 8, ll. 3. See also Col. 9, ll. 12-19 (“A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard. The space needed for the keyboard or keypad is quite small, and will not completely obscure the work area of the spreadsheet, word processor, etc., which is the active application program.”).</p>
1.6	providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.	<p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code. The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface.” Abstract.</p> <p>“The alternative input subsystem module provides communication between the attached user friendly input devices and the remainder of the advanced user interface as well as application programs through the integrated operating environment. The alternative input subsystem module receives the input signals generated by the input devices and translates them to input messages useable in the advanced user interface.” Col. 3, ll. 65-68 to Col. 4, ll. 1-5.</p> <p>The AUI manages the selection of the input method and communicates the input data</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³ and/or U.S. Patent No. 5,946,499⁴
		of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs. Col. 8, ll. 3.
		“After receiving an input message from Presentation Manager.TM. 156 together with information about which is the active application program, the PM-Link 201 refers back to the alternate input subsystem 203 to determine whether the “keyboard” or “mouse” message it received is in fact a keyboard or a mouse message, or rather a touch, voice, image, or other message. The PM-Link 201 then passes the true message to the appropriate application program.” Col. 14, ll. 7-16.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
2	2. The method of claim 1 wherein providing the input to the computer program comprises communicating information representative of the input to a graphical windowing environment.	“As shown [in Figure 3], the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.” Col. 8, ll. 65-68 to Col. 9, ll. 1-2.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
-------	-----------------------	---

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
3	3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.	<p>The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs.” Col. 8, ll. 3.</p> <p>“The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract.</p> <p>“The alternative input subsystem module provides communication between the attached user friendly input devices and the remainder of the advanced user interface as well as application programs through the integrated operating environment. The alternative input subsystem module receives the input signals generated by the input devices and translates them to input messages useable in the advanced user interface.” Col. 3, ll. 65-68 to Col. 4, ll. 1-5.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
4	4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same	<p>The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs.” Col. 8, ll. 3.</p> <p>“Another recent trend is to provide some sort of integration of computer program applications. Without integration, the user must employ separate application programs for word processing, database manipulation, graphics and electronic mail functions,</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
	manner as if the input was received via a hardware keyboard.	and so forth. It is often quite difficult to integrate the outputs of the different programs into a single desired output. One solution has been to write a single integrated piece of software which incorporates a variety of applications which is called a multiple-function program.” Col. 2, ll. 43-52. “The interface profile module 104 is comprised of sets of application profiles 105 and the user profiles 107, which are files which list input messages produced by the AIS 103 from the input signals received by input devices 36, 38, 40, mapped to keyboard, mouse or other commands which are usable by existing application programs, e.g., mouse clicks, keystroke messages, MACROs, utility programs, etc.” Col. 8, ll. 55-62.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving information corresponding to a keyboard character entered via the displayed keyboard.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 9, ll. 9-16.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
8	8. The method of claim 1 further comprising, hiding the input panel.	<p>See Claims 1.1 – 1.6.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include . . . an image magnifying utility . . . The image magnifying utility will magnify a rectangle of fixed size around a point at which the appropriate gesture was made. The utility allows very accurate positioning of a cursor in the expanded image. After stylus liftoff, the normal size display is restored, and the selected cursor coordinates are sent to the active application program.” Col. 9, ll. 3-25.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and U.S. Patent No. 5,148,155 (“Martin ‘155”) renders this claim obvious. Martin ‘155 teaches the hiding of an input panel. See Martin ‘155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
9	9. The method of claim 1 further comprising, docking the input panel.	See Claim 23 of Tannenbaum ‘951 (“The computer system as recited in claim 22, which further comprises a touch input device disposed over the viewing surface of the

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		display device, and wherein the advanced user interface further comprises a pop-up keyboard module which displays an image of a keyboard on the display device [.] ” (emphasis added). To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2. In addition, the combination of Tannenbaum ‘951 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
10	10. At least one computer-readable medium having computer-executable instructions, which when executed perform the method of claim 1.	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 6, ll. 40-51.
Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 6, ll. 40-51.
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each executable input method is an interchangeable software component distinct from one or more application programs, each executable input method having a defined interface set such that the executable input method is connectable to the application programs;	See Claims 1.1 – 1.6. The invention provides for multiple input types which supply user input to the computer system. See Figure 3. The input methods are each separately executable from the application programs. Col. 8, ll. 65-67 (“As shown, the AUI utility 109 can be written as a separate application program[.]”). The input methods are connectable to the application programs through an advanced user interface and/or the operating system. Col. 1, ll. 15-19 (this invention “relates to an advanced user interface which allows a user to select one or more input devices to input data into a computer running a program originally written for a different input

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		device in a multiapplication environment”). See also Figure 3. “The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding environment or other useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract.
11.2	opening an input window on a display of the computer system independent of a window of an active application program; and	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12. The AUI Utility is independent of the application program. Col. 8, ll. 8-13 Col. 8, ll. 65-68 to Col. 9, ll. 1-2 (“the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.”).
11.3	displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input received by the selected executable input method via the interactive input panel is provided to the active application program as if the information was received via user input at a hardware input device.	See Claims 1.1 – 1.6. “A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 3-12 (emphasis added). The input methods are each separately executable from the application programs. Col. 8, ll. 65-67 (“As shown, the AUI utility 109 can be written as a separate application program[.]”). “After receiving an input message from Presentation Manager.TM. 156 together with information about which is the active application program, the PM-Link 201 refers back to the alternate input subsystem 203 to determine whether the “keyboard” or

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		“mouse” message it received is in fact a keyboard or a mouse message, or rather a touch, voice, image, or other message. The PM-Link 201 then passes the true message to the appropriate application program.” Col. 14, ll. 7-16.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
13	13. The computer-readable medium of claim 11 further comprising, providing a Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuable to display a selectable list of the plurality of executable input methods.	<p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract.</p> <p>“It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 38-40.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 3-12 (emphasis added).</p> <p>The input methods are each separately executable from the application programs. Col. 8, ll. 65-67 (“As shown, the AUI utility 109 can be written as a separate application program[.]”).</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
14	14. The computer-readable medium of claim 13 further comprising, receiving a selection of one of the plurality of executable input methods displayed in the list as a selected executable input method, and in response, closing any open input window, and opening a new input window corresponding to the selected executable input method.	See Claim 8. “A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 3-12 (emphasis added).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
15	15. At least one computer-readable medium having computer-executable instructions, which when executed perform steps, comprising:	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 6, ll. 40-51.
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	See Claims 1.1 – 1.6. “A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 3-12 (emphasis added). The AUI Utility is independent of the application program. Col. 8, ll. 8-13 Col. 8, ll. 65-68 to Col. 9, ll. 1-2 (“the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.”). To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement an actuatable icon representative of an input method list. For example, the invention discloses several types of interface utilities 109, and it would have been obvious to one of skill in the art to include this step to permit the user to select one of those input utilities. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.
15.2	invoking a selected input method in response to a user selecting an icon corresponding to the selected input method, including presenting an input panel window; and	See Claims 1.1 – 1.6. “By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 9, ll. 9-16.
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	See Claims 15.2.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
16	16. The computer-readable medium of claim 15 wherein accepting user data includes detecting user interaction with a touch-sensitive display.	<p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 9, ll. 9-16.</p> <p>See Claim 16 of Tannenbaum ‘951 (“The computer system as recited in claim 13, which further comprises a gesture recognition unit which interprets the input signals transmitted by a touch input device in response to a user drawing symbols on a surface detected by the touch input device.”).</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
17	17. The computer-readable medium of claim 15 wherein each input method comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Tannenbaum ‘951 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u></p>
		<p>independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p><u>The combination of Tannenbaum ‘951 and U.S. Patent No. 5,946,499 (“the Saunders Reference”) also renders this claim obvious. In the Saunders Reference, text input methods communicate with the text services manager (“TSM”) through an object model interface. See Saunders Reference, Col. 4, ll. 23-43 (“Each second data structure TSMContext 40 is suitably attached to the first data structure TSMDocument 41 and contains two pieces of information, a LocaleObject 42 and a SOM (system object model) object 44. A LocaleObject 42 appropriately represents a text service and supplies the TSM 20 with the information necessary to load that text service, including a text service class used by the TSM 20 to access the text service and a SOM class name for the SOM class implemented by the text service, as is well understood by those skilled in the art. A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter</u></p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		<u>(component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</u> Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	“The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value. Col. 3, ll. 21-

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		<p>35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and U.S. Patent No. 5,454,046 (“the Carmen Reference”)⁵ renders this claim obvious. See Carmen Reference, Col. 10, ll. 14-22 (“The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation is used instead of ASCII or Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all languages of the world, whereas ASCII and ANSI representations are limited to 256 characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>See Figure 3A of Carmen Reference.</p>

⁵ “Universal Symbolic Handwriting Recognition System” issued to Carman; filed September 17, 1993; issued September 26, 1995.

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499</p>
		<div data-bbox="324 478 756 989"> </div> <p style="text-align: center;"><i>FIG. 3A</i></p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>Similarly, the combination of Tannenbaum ‘951 and U.S. Patent No. 5,455,901 (“the Friend Reference”)⁶ renders this claim obvious. See Claim 6 of Friend Reference (“The device of claim 5 wherein said translator comprises a generator for generating machine readable characters from said strokes.”); Claim 7 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise Unicode characters.”); Claim 8 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise ASCII characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field</p>

⁶ “Input Device With Deferred Translation” issued to Friend; filed September 12, 1994; issued October 3, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
		and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
19	19. In a computing environment, a system comprising,	“An advanced user interface for use with a computer system operating on an integrated operating environment.” Abstract.
19.1	a manager component stored on one or more computer-readable media and configured: to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable software component distinct from one or more computer programs, and	“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract. “It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 38-40. The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. <i>See</i> Fig. 3. The input methods are each separately executable from the application programs. Col. 8, ll. 65-68 to Col. 9, ll. 1-2 (“the AUI utility 109 can be written as a separate

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
	to send input data corresponding to a user input received at the selected input method to a graphical windowing environment; and	application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.”). The AUI has “translation capabilities” for sending user data to the application programs. Col. 8, ll. 3.
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	The Integrated Operating Environment (windowing environment) serves as the interface between the AUI and the applications. See Figures 3 & 6; see also Abstract (“The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.”). The AUI Utility is independent of the application program. Col. 8, ll. 8-13 Col. 8, ll. 65-68 to Col. 9, ll. 1-2 (“the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	“The integrated operating environment allows a plurality of application programs to be running simultaneously, one of which is designated the active application program to which all input data is directed.” Abstract.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
21	21. The system of claim 19 further comprising an input panel window corresponding to the selected input method.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
23	23. The system of claim 21 wherein the manager component selectively displays and hides the input panel window.	See Claim 8. “By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
25	25. The system of claim 19 where the input method is displayed on a touch-sensitive display screen.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.
Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
26	26. The system of claim 19 wherein the manager component transfers information from the computer program to the selected input method.	See Figure 8. “After being recognized by the gesture recognition unit 213 as a circle gesture, the PM-Link 201 passes the circle gesture to the spreadsheet 150. The spreadsheet returns an “R0” message at 285, indicating that the circle gesture was not understood. Since the circle gesture was not understood at 287, the PM-Link 201 refers to the application profile 205 for the spreadsheet 150 at 289 and finds no corresponding command for the circle gesture. The PM-Link 201 then refers to the user profile 207 at 291 which contains the corresponding command “invoke the pop-up keyboard”. As the command from the user profile 207 has the highest priority at 293, the PM-Link 201 sends this message to the AUI utilities module 209 at 295 and the pop-up keyboard is presented to the user on the display.” Col. 17, ll. 8-22.
Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined interface set.	See Figure 4. See Claim 10 of Tannenbaum ‘951 (“The advanced user interface as recited in claim 9 wherein the interface profiles contain mappings of touch input messages against corresponding mouse commands.”). “After determining the active application program, spreadsheet 50 at 119, the environment link 101 refers to the application profile 105 of the spreadsheet 50 for the command which corresponds to the input message “GRAPH” which will be recognized by the spreadsheet 50, i.e., the menu selection-- graph--.” Col. 9, ll. 48-54.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 and/or U.S. Patent No. 5,946,499
28	28. The system of claim 19 wherein the selected input method comprises an object.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”). In addition, the combination of Tannenbaum ‘951 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u></p>
		<p>independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p><u>The combination of Tannenbaum ‘951 and U.S. Patent No. 5,946,499 (“the Saunders Reference”) also renders this claim obvious. In the Saunders Reference, text input methods communicate with the text services manager (“TSM”) through an object model interface. See Saunders Reference, Col. 4, ll. 23-43 (“Each second data structure TSMContext 40 is suitably attached to the first data structure TSMDocument 41 and contains two pieces of information, a LocaleObject 42 and a SOM (system object model) object 44. A LocaleObject 42 appropriately represents a text service and supplies the TSM 20 with the information necessary to load that text service, including a text service class used by the TSM 20 to access the text service and a SOM class name for the SOM class implemented by the text service, as is well understood by those skilled in the art. A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter</u></p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		<u>(component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</u> Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
29	29. The system of claim 19 wherein the selected input method draws an input panel in an input panel window displayed in the graphical windowing environment.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
30	30. The system of claim 29 wherein the manager component selectively displays	See Claim 8. “By issuing the proper command, e.g., a circle gesture or touching a keyboard icon,

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
	and hides the display of the input panel window.	the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
31	31. The system of claim 29 wherein the manager component docks the input panel window.	<p>See Claim 9.</p> <p>See Claim 23 of Tannenbaum ‘951 (“The computer system as recited in claim 22, which further comprises a touch input device disposed over the viewing surface of the display device, and wherein the advanced user interface further comprises a pop-up keyboard module which displays an image of a keyboard on the display device [.]”) (emphasis added).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified,</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 <u>and/or U.S. Patent No. 5,946,499</u>
		predictable solutions.

EXHIBIT L

Exemplar Chart of U.S. Patent 7,411,582**U.S. Patent 5,946,499 (“Saunders ‘499’”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

Claim	U.S. Patent 7,411,582	U.S. Patent 5,946,499 (“Saunders ‘499’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
1	1. In a computing environment, a computer-implemented method comprising:	<p>“A method, and system for supporting a plurality of textual manipulations and inputs in a computer system transparently to application programs running on the computer system, with the computer system including a text services manager, are described.” Abstract.</p> <p>“The present invention provides system and method aspects to support a plurality of textual manipulations and inputs and provide appropriate textual outputs. In one system aspect, the system includes at least one application program, and an operating system supporting operations of the at least one application program, the operating system comprising a text services manager for receiving data including text input</p>

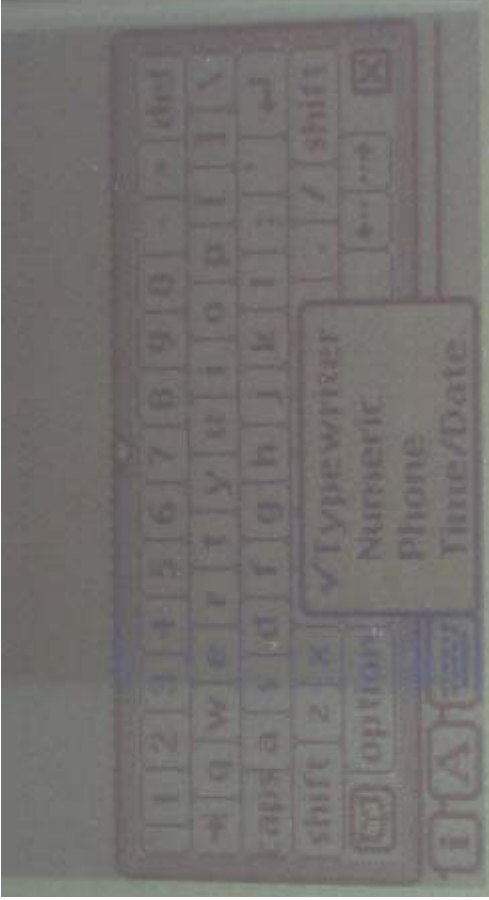
NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Method And Apparatus For Processing Text Inputs From Multiple Input Devices In A Plurality Of Applications” issued to Saunders; filed May 10, 1996; issued August 31, 1999.

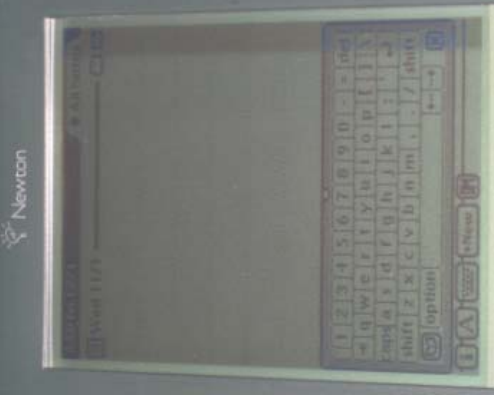
² Kraig Brockschmidt, Inside OLE (2d ed. 1995).

³ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,946,499 (“Saunders ‘499”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
		events and linguistic aid events from a plurality of text input device formats through the at least one application program and providing a plurality of text service event handlers in response to the received data.” Col. 1, ll. 55-65.
1.1	displaying an actuatable icon representative of an input method list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a computer-executable software component distinct from the computer programs;	<p>“An application program 14 preferably interacts with the TSM 20 with the creation of a data structure (TSM Document) i.e., an opaque structure in memory that organizes related pieces of information, the basic types of which include files, lists, arrays, records, trees, and tables, as is well understood by those skilled in the art. The application program 14 further preferably performs operations, such as activate and deactivate, on the data structure, <i>where activate suitably refers to activation of an input method and insertion of a menu for the TSM 20 into the application program 14 menu bar</i>, while deactivate suitably refers to deactivation of an input method and removal of the TSM’s menu from the menu bar for the application program 14.” Col. 3, ll. 53-65 (emphasis added).</p> <p>“A method, and system for supporting a plurality of textual manipulations and inputs in a computer system transparently to application programs running on the computer system[.]” Abstract.</p> <p>“[T]he text input device formats include, but are not limited to, keyboard device formats, mouse device formats, stylus device formats, and microphone device formats.” Col. 1, ll. 66-67 to Col. 2, ll. 1-2.</p> <p>“With the present invention, handling textual inputs from a variety of input devices readily occurs in different types of application programs processing text.” Col. 2, ll. 40-42.</p> <p>“The present invention relates to formation and transformation of text inputs from a variety of input devices for a plurality of application programs.” Col. 2, ll. 66-67 to Col. 3, ll. 1.</p> <p>The various input methods are executable independent of the application programs.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,946,499 (“Saunders ‘499”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
		<p>Col. 2, ll. 40-42 (“With the present invention, handling textual inputs from a variety of input devices readily occurs in different types of application programs processing text.”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to display an actuatable icon representative of an input method list to the user. For example, Saunders ‘499 discloses several input methods, and it would have been obvious to one of skill in the art to display an actuatable icon representative of the input method list. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders ‘499 and the MessagePad 120 renders this claim obvious. As shown below, the MessagePad 120 teaches the display of an actuatable icon representative of an input method list to the user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> 

Claim	U.S. Patent 7,411,582	U.S. Patent 5,946,499 (“Saunders ‘499”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
		Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.2	in response to actuation of the actuatable icon, displaying the input method list;	See Claim 1.1.
1.3	receiving a selection of an input method from the input method list;	<p>See Claim 1.1. To the extent not expressly or inherently disclosed, it would have been obvious to permit the user to select a particular input method. For example, Saunders ‘499 discloses several input methods, and it would have been obvious to one of skill in the art to permit selection of a particular input method by the user. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders ‘499 and the MessagePad 120 renders this claim obvious. As shown below, the MessagePad 120 permits the user to select a particular input method, in this case, the on-screen keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,946,499 (“Saunders ‘499”) alone and/or in combination with U.S. Patent No. 5,455,901 and/or U.S. Patent 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
		 <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	See Claim 1.3.
1.5	receiving input via the interactive input panel; and	<p>“In one system aspect, the system includes at least one application program, and an operating system supporting operations of the at least one application program, the operating system comprising a text services manager for receiving data including text input events and linguistic aid events from a plurality of text input device formats through the at least one application program and providing a plurality of text service event handlers in response to the received data.” Col. 1, ll. 57-65 (emphasis added).</p> <p>To the extent not expressly or inherently disclosed, it would have been an obvious step</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,946,499 (“Saunders ‘499”) alone and/or in combination with U.S. Patent No. 5,455,901 and/or U.S. Patent 5,148,155 and/or Brockschmidt² and/or the MessagePad 120³</p>
		<p>to permit input to be received via the interactive input panel. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders ‘499 and the MessagePad 120 renders this claim obvious. As shown below, the MessagePad 120 allows text input to be received via the interactive input panel. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <div data-bbox="690 743 1344 1253" data-label="Image"> </div> <div data-bbox="690 205 1344 707" data-label="Image"> </div> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,946,499 (“Saunders ‘499”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
		and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.6	providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.	<p>“Preferably, the application program 14 is unaware of the text services receiving and handling the events.” Col. 4, ll. 46-48.</p> <p>“The handling of events as described with reference to FIGS. 4a and 4b preferably occurs transparently to the application program 14. However, a protocol suitably exists to allow communication between a particular text service and the application to facilitate event handling.” Col. 6, ll. 10-15.</p> <p>“The present invention relates to formation and transformation of text inputs from a variety of input devices for a plurality of application programs.” Col. 2, ll. 66-67 to Col. 3, ll. 1.</p>

Claim	U.S. Patent 7,411,582	
2	2. The method of claim 1 wherein providing the input to the computer program comprises communicating information representative of the input to a graphical windowing environment.	<p>See Figure 1.</p> <p>“The present invention provides system and method aspects to support a plurality of textual manipulations and inputs and provide appropriate textual outputs. In one system aspect, the system includes at least one application program, and an operating system supporting operations of the at least one application program, the operating system comprising a text services manager for receiving data including text input events and linguistic aid events from a plurality of text input device formats through the at least one application program and providing a plurality of text service event handlers in response to the received data.” Col. 1, ll. 55-65.</p>

Claim 3	U.S. Patent 7,411,582	<p>3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.</p>	<p>“As shown, upon the occurrence of an event to be handled by a text service, an application program 14 makes a call for a new data structure, e.g., TSMDocument, from the TSM 20 (step 30). In turn, the data structure TSMDocument is suitably created (step 32). Preferably, the data structure provides a high level interface to text services, and gives the application program 14 access to in-line processing of inputs.” Col 1, ll. 1-8.</p>
------------	-----------------------	--	---

Claim 4	U.S. Patent 7,411,582	<p>4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same manner as if the input was received via a hardware keyboard.</p>	<p>See Figure 1.</p> <p>“The present invention provides system and method aspects to support a plurality of textual manipulations and inputs and provide appropriate textual outputs. In one system aspect, the system includes at least one application program, and an operating system supporting operations of the at least one application program, the operating system comprising a text services manager for receiving data including text input events and linguistic aid events from a plurality of text input device formats through the at least one application program and providing a plurality of text service event handlers in response to the received data.” Col. 1, ll. 55-65.</p> <p>“Preferably, the application program 14 is unaware of the text services receiving and handling the events.” Col. 4, ll. 46-48.</p> <p>“The handling of events as described with reference to FIGS. 4a and 4b preferably occurs transparently to the application program 14. However, a protocol suitably exists to allow communication between a particular text service and the application to facilitate event handling.” Col. 6, ll. 10-15.</p>
------------	-----------------------	--	---

Claim	U.S. Patent 7,411,582	
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving information corresponding to a keyboard character entered via the displayed keyboard.	<p>“Events to be handled include, but are not limited to, key events, such as keystrokes on a keyboard, mouse events, such as mouse movements or tablet inputs, window events, such as events concerned with the user interface, and speech events, such as sound from a microphone” Col. 4, ll. 66-67 to Col. 5, ll. 1-4.</p> <p>“For each input method, both keyboard and non-keyboard based, a text service preferably acts as a filter for input device events that are passed to an application program 14, thus intercepting entries of text and interacting with the user and application program 14 to convert raw events into a desired text stream.” Col. 3, ll. 35-40.</p>

Claim	U.S. Patent 7,411,582	
8	8. The method of claim 1 further comprising, hiding the input panel.	<p>See Claims 1.1 – 1.6.</p> <p>“An application program 14 preferably interacts with the TSM 20 with the creation of a data structure (TSMDocument) i.e., an opaque structure in memory that organizes related pieces of information, the basic types of which include files, lists, arrays, records, trees, and tables, as is well understood by those skilled in the art. The application program 14 further preferably performs operations, such as activate and deactivate, on the data structure, where activate suitably refers to activation of an input method and insertion of a menu for the TSM 20 into the application program 14 menu bar, while deactivate suitably refers to deactivation of an input method and removal of the TSM's menu from the menu bar for the application program 14.” Col. 3, ll. 53-65.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders ‘499 and U.S. Patent No. 5,148,155 (“Martin ‘155”) renders this claim obvious. Martin ‘155 teaches the hiding of an input panel. See Martin ‘155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be</p>

Claim	U.S. Patent 7,411,582	<p>motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
-------	-----------------------	--


Claim	U.S. Patent 7,411,582	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders '499 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
-------	-----------------------	--

Claim	U.S. Patent 7,411,582	
10	10. At least one computer-readable	"A method, and system for supporting a plurality of textual manipulations and inputs

Claim	U.S. Patent 7,411,582	
	medium having computer-executable instructions, which when executed perform the method of claim 1.	in a computer system transparently to application programs running on the computer system, with the computer system including a text services manager, are described.” Abstract. “Although the present invention has been described in accordance with the embodiments shown, one of ordinary skill in the art will recognize that there could be variations to the embodiment and those variations would be within the spirit and scope of the present invention. For example, the TSM and operating system could be implemented from programming instructions stored on a computer readable medium, such as a floppy disk. Accordingly, many modifications may be made by one of ordinary skill without departing from the spirit and scope of the present invention, the scope of which is defined by the following claims.” Col. 7, ll. 44-54.

Claim	U.S. Patent 7,411,582	
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	“A method, and system for supporting a plurality of textual manipulations and inputs in a computer system transparently to application programs running on the computer system, with the computer system including a text services manager, are described.” Abstract. “Although the present invention has been described in accordance with the embodiments shown, one of ordinary skill in the art will recognize that there could be variations to the embodiment and those variations would be within the spirit and scope of the present invention. For example, the TSM and operating system could be implemented from programming instructions stored on a computer readable medium, such as a floppy disk. Accordingly, many modifications may be made by one of ordinary skill without departing from the spirit and scope of the present invention, the scope of which is defined by the following claims.” Col. 7, ll. 44-54.
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each	“[T]he text input device formats include, but are not limited to, keyboard device formats, mouse device formats, stylus device formats, and microphone device formats.” Col. 1, ll. 66-67 to Col. 2, ll. 1-2.

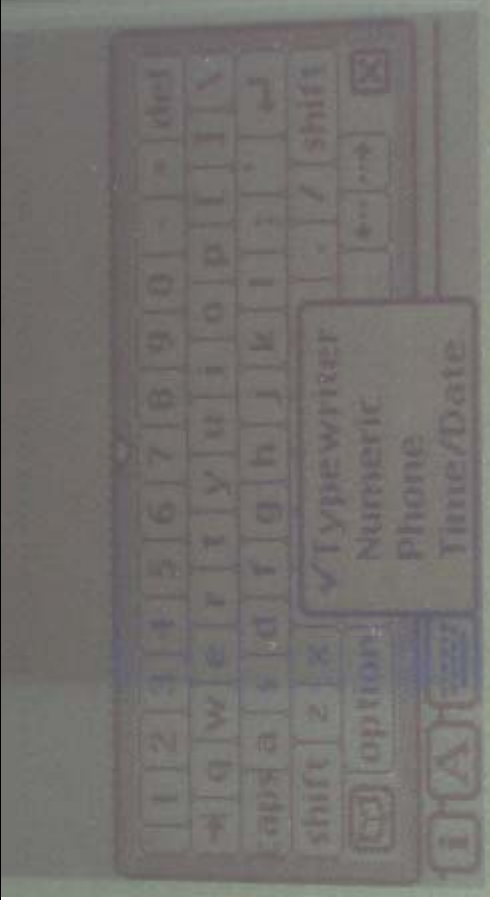
Claim	U.S. Patent 7,411,582	executable input method is an interchangeable software component distinct from one or more application programs, each executable input method having a defined interface set such that the executable input method is connectable to the application programs;	
11.2	opening an input window on a display of the computer system independent of a window of an active application program;	<p>“With the present invention, handling textual inputs from a variety of input devices readily occurs in different types of application programs processing text.” Col. 2, ll. 40-42.</p> <p>“The present invention relates to formation and transformation of text inputs from a variety of input devices for a plurality of application programs.” Col. 2, ll. 66-67 to Col. 3, ll. 1.</p> <p>The various input methods are executable independent of the application programs. Col. 2, ll. 40-42 (“With the present invention, handling textual inputs from a variety of input devices readily occurs in different types of application programs processing text.”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system capable of calling functions in the manager component via a defined interface set. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Saunders ‘499 and U.S. Patent 5,157,384 (“Greanias ‘384”) renders this claim obvious. Greanias ‘384 teaches calling functions in the manager component via a defined interface set. See Greanias ‘384 Claim 37 and Col. 8, ll. 59-65. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to open an input window independent of the application window. Col. 7, ll. 44-54.</p>	

Claim	U.S. Patent 7,411,582	
	and	<p>In addition, the combination of Saunders '499 and the MessagePad 120 renders this claim obvious. As shown below, the MessagePad 120 opens an input window independent of the application window. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p>  <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
11.3	displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input received by the selected executable input	<p>See Claim 11.2.</p> <p>“The present invention relates to formation and transformation of text inputs from a variety of input devices for a plurality of application programs.” Col. 2, ll. 66-67 to Col. 3, ll. 1.</p>



Claim	U.S. Patent 7,411,582	
	method via the interactive input panel is provided to the active application program as if the information was received via user input at a hardware input device.	<p>The various input methods are executable independent of the application programs. Col. 2, ll. 40-42 (“With the present invention, handling textual inputs from a variety of input devices readily occurs in different types of application programs processing text.”).</p> <p>“Preferably, the application program 14 is unaware of the text services receiving and handling the events.” Col. 4, ll. 46-48.</p> <p>“The handling of events as described with reference to FIGS. 4a and 4b preferably occurs transparently to the application program 14. However, a protocol suitably exists to allow communication between a particular text service and the application to facilitate event handling.” Col. 6, ll. 10-15.</p>

Claim	U.S. Patent 7,411,582	
13	13. The computer-readable medium of claim 11 further comprising, providing a Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuable to display a selectable list of the plurality of executable input methods.	<p>“An application program 14 preferably interacts with the TSM 20 with the creation of a data structure (TSMDocument) i.e., an opaque structure in memory that organizes related pieces of information, the basic types of which include files, lists, arrays, records, trees, and tables, as is well understood by those skilled in the art. The application program 14 further preferably performs operations, such as activate and deactivate, on the data structure, <i>where activate suitably refers to activation of an input method and insertion of a menu for the TSM 20 into the application program 14 menu bar</i>, while deactivate suitably refers to deactivation of an input method and removal of the TSM's menu from the menu bar for the application program 14.” Col. 3, ll. 53-65 (emphasis added).</p> <p>“A method, and system for supporting a plurality of textual manipulations and inputs in a computer system transparently to application programs running on the computer system[.]” Abstract.</p> <p>“[T]he text input device formats include, but are not limited to, keyboard device</p>

Claim	U.S. Patent 7,411,582	<p>formats, mouse device formats, stylus device formats, and microphone device formats.” Col. 1, ll. 66-67 to Col. 2, ll. 1-2.</p> <p>“With the present invention, handling textual inputs from a variety of input devices readily occurs in different types of application programs processing text.” Col. 2, ll. 40-42.</p> <p>“The present invention relates to formation and transformation of text inputs from a variety of input devices for a plurality of application programs.” Col. 2, ll. 66-67 to Col. 3, ll. 1.</p> <p>The various input methods are executable independent of the application programs. Col. 2, ll. 40-42 (“With the present invention, handling textual inputs from a variety of input devices readily occurs in different types of application programs processing text.”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to display an actuable menu button representative of an input method list to the user. For example, Saunders ‘499 discloses several input methods, and it would have been obvious to one of skill in the art to display an actuable menu button representative of the input method list. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders ‘499 and the MessagePad 120 renders this claim obvious. As shown below, the MessagePad 120 teaches the display of an actuable menu button representative of an input method list to the user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p>
-------	-----------------------	---

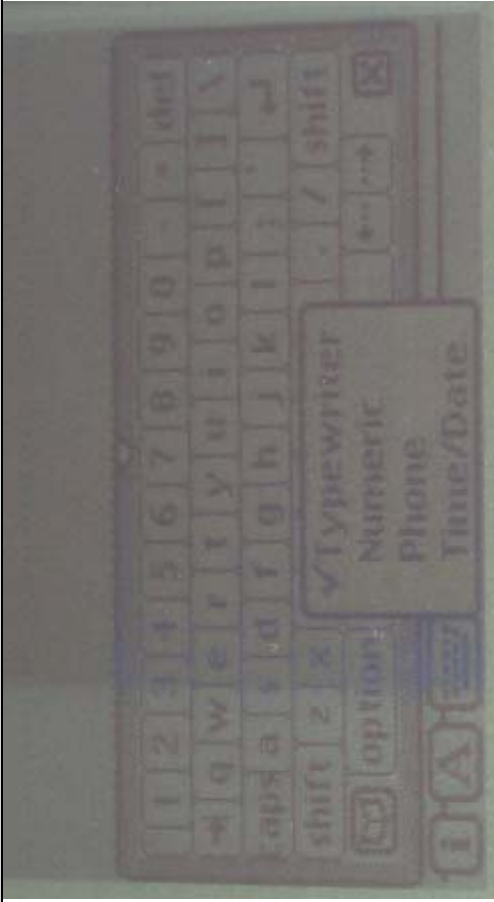
Claim	U.S. Patent 7,411,582	 <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
-------	-----------------------	--


Claim	U.S. Patent 7,411,582	<p>See Claim 1.1. To the extent not expressly or inherently disclosed, it would have been obvious to permit the user to select a particular input method. For example, Saunders '499 discloses several input methods, and it would have been obvious to one of skill in the art to permit selection of a particular input method by the user. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders '499 and the MessagePad 120 renders this claim obvious. As shown below, when the user selects the Numeric input method option on the MessagePad 120, the Typewriter input method window closes, and the Numeric input method window opens.</p>
-------	-----------------------	---

Claim	U.S. Patent 7,411,582	
	<div data-bbox="212 407 527 1050">  </div> <div data-bbox="565 434 885 1022">  </div> <p data-bbox="922 170 1105 1287">One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p data-bbox="1143 170 1289 1287">Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>	

Claim	U.S. Patent 7,411,582	
15	15. At least one computer-readable medium having computer-executable instructions, which when executed perform steps, comprising:	<p>“A method, and system for supporting a plurality of textual manipulations and inputs in a computer system transparently to application programs running on the computer system, with the computer system including a text services manager, are described.” Abstract.</p> <p>“Although the present invention has been described in accordance with the embodiments shown, one of ordinary skill in the art will recognize that there could be variations to the embodiment and those variations would be within the spirit and scope of the present invention. For example, the TSM and operating system could be implemented from programming instructions stored on a computer readable medium, such as a floppy disk. Accordingly, many modifications may be made by one of ordinary skill without departing from the spirit and scope of the present invention, the scope of which is defined by the following claims.” Col. 7, ll. 44-54.</p>
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	<p>“An application program 14 preferably interacts with the TSM 20 with the creation of a data structure (TSMDocument) i.e., an opaque structure in memory that organizes related pieces of information, the basic types of which include files, lists, arrays, records, trees, and tables, as is well understood by those skilled in the art. The application program 14 further preferably performs operations, such as activate and deactivate, on the data structure, <i>where activate suitably refers to activation of an input method and insertion of a menu for the TSM 20 into the application program 14 menu bar</i>, while deactivate suitably refers to deactivation of an input method and removal of the TSM's menu from the menu bar for the application program 14.” Col. 3, ll. 53-65 (emphasis added).</p> <p>“A method, and system for supporting a plurality of textual manipulations and inputs in a computer system transparently to application programs running on the computer system[.]” Abstract.</p> <p>“[T]he text input device formats include, but are not limited to, keyboard device formats, mouse device formats, stylus device formats, and microphone device formats.” Col. 1, ll. 66-67 to Col. 2, ll. 1-2.</p> <p>“With the present invention, handling textual inputs from a variety of input devices</p>

Claim	U.S. Patent 7,411,582	<p>readily occurs in different types of application programs processing text.” Col. 2, ll. 40-42.</p> <p>“The present invention relates to formation and transformation of text inputs from a variety of input devices for a plurality of application programs.” Col. 2, ll. 66-67 to Col. 3, ll. 1.</p> <p>The various input methods are executable independent of the application programs. Col. 2, ll. 40-42 (“With the present invention, handling textual inputs from a variety of input devices readily occurs in different types of application programs processing text.”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to display icons representative of an input method list to the user. For example, Saunders ‘499 discloses several input methods, and it would have been obvious to one of skill in the art to present icons corresponding to a plurality of input methods available for a computer application. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders ‘499 and the MessagePad 120 renders this claim obvious. As shown below, the MessagePad 120 teaches the display of an actuatable icon representative of an input method list to the user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p>
-------	-----------------------	---

Claim	U.S. Patent 7,411,582		
15.2	invoking a selected input method in response to a user selecting an icon corresponding to the selected input method, including presenting an input panel window; and	 <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>	<p>To the extent not expressly or inherently disclosed, it would have been obvious to permit the user to select a particular input method. For example, Saunders '499 discloses several input methods, and it would have been obvious to one of skill in the art to permit selection of a particular input method by the user. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders '499 and the MessagePad 120 renders this claim obvious. As shown below, the MessagePad 120 permits the user to select a particular input method, in this case, the on-screen keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p>

Claim	U.S. Patent 7,411,582	
		 <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	See Claims 1.3 – 1.6.

Claim	U.S. Patent 7,411,582	
16	16. The computer-readable medium of claim 15 wherein accepting user data includes detecting user interaction with a touch-sensitive display.	“Processing of text inputs into a computer system has many facets. Application programs typically handle keystroke entries from a keyboard, as well as selection of characters from a keyboard palette displayed on a display device through the use of a mouse or other pointing device. On-going technological improvements to computer

Claim	U.S. Patent 7,411,582	
		systems have enhanced the methods for inputting data, including text, into computers. Such improvements include the ability to input text as handwriting, via a pen or stylus device. Another improvement includes the ability to input text captured as speech, via a microphone input device.” Col. 1, ll. 12-22.

Claim	U.S. Patent 7,411,582	
17	17. The computer-readable medium of claim 15 wherein each input method comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.	<p>“As shown, upon the occurrence of an event to be handled by a text service, an application program 14 makes a call for a new data structure, e.g., TSMDocument, from the TSM 20 (step 30). In turn, the data structure TSMDocument is suitably created (step 32). Preferably, the data structure provides a high level interface to text services, and gives the application program 14 access to in-line processing of inputs.” Col 1, ll. 1-8.</p> <p>“A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.” Col. 4, ll. 33-44.</p>

Claim	U.S. Patent 7,411,582	
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value. Col. 7, ll. 44-54.

Claim	U.S. Patent 7,411,582	<p>In addition, the combination of Saunders '499 and U.S. Patent No. 5,454,046 ("the Carman Reference") renders this claim obvious. See Carman Reference, Col. 10, ll. 14-22 ("The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation defined by the Unicode Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all languages of the world, whereas ASCII and ANSI representations are limited to 256 characters."). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>See Figure 3A of Carman Reference.</p> <div><p>FIG. 3A</p></div> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified,</p>
-------	-----------------------	--

Claim	U.S. Patent 7,411,582	<p>predictable solutions.</p> <p>Similarly, the combination of Saunders '499 and U.S. Patent No. 5,455,901 ("the Friend Reference") renders this claim obvious. See Claim 6 of Friend Reference ("The device of claim 5 wherein said translator comprises a generator for generating machine readable characters from said strokes."); Claim 7 of Friend Reference ("The device of claim 6 wherein said machine readable characters comprise Unicode characters."); Claim 8 of Friend Reference ("The device of claim 6 wherein said machine readable characters comprise ASCII characters."). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
-------	-----------------------	---

Claim	U.S. Patent 6,339,780	<p>"A method, and system for supporting a plurality of textual manipulations and inputs in a computer system transparently to application programs running on the computer system, with the computer system including a text services manager, are described."</p> <p>Abstract.</p> <p>"The present invention provides system and method aspects to support a plurality of textual manipulations and inputs and provide appropriate textual outputs. In one system aspect, the system includes at least one application program, and an operating system supporting operations of the at least one application program, the operating system comprising a text services manager for receiving data including text input events and linguistic aid events from a plurality of text input device formats through</p>
-------	-----------------------	--

Claim	U.S. Patent 6,339,780	
19.1	<p>a manager component stored on one or more computer-readable media and configured:</p> <p>to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable software component distinct from one or more computer programs, and</p> <p>to send input data corresponding to a user input received at the selected input method to a graphical windowing environment; and</p>	<p>the at least one application program and providing a plurality of text service event handlers in response to the received data.” Col. 1, ll. 55-65.</p> <p>“In one system aspect, the system includes at least one application program, and an operating system supporting operations of the at least one application program, the operating system comprising a <i>text services manager</i> for receiving data including text input events and linguistic aid events from a plurality of text input device formats through the at least one application program and providing a plurality of text service event handlers in response to the received data.” Col. 1, ll. 57-65 (emphasis added).</p> <p>“Events to be handled include, but are not limited to, key events, such as keystrokes on a keyboard, mouse events, such as mouse movements or tablet inputs, window events, such as events concerned with the user interface, and speech events, such as sound from a microphone” Col. 4, ll. 66-67 to Col. 5, ll. 1-4.</p> <p>“The coordination of text input processing suitably occurs with a text services manager (TSM) 20 included in the operating system of computer 10.” Col. 3, ll. 23-26.</p> <p>“The TSM 20 preferably controls interactions between text services and the application program 14.” Col. 3, ll. 31-32.</p> <p>“For purposes of this discussion, text services preferably refer to a class of functions that form or transform text.” Col. 3, ll. 32-34.</p> <p>“For each input method, both keyboard and non-keyboard based, a text service preferably acts as a filter for input device events that are passed to an application program 14, thus intercepting entries of text and interacting with the user and application program 14 to convert raw events into a desired text stream.” Col. 3, ll. 35-40.</p> <p>The various input methods are executable independent of the application programs. Col. 2, ll. 40-42 (“With the present invention, handling textual inputs from a variety of input devices readily occurs in different types of application programs processing</p>

Claim	U.S. Patent 6,339,780	text.”).
		“In one system aspect, the system includes at least one application program, and an operating system supporting operations of the at least one application program, the operating system comprising a text services manager for receiving data including text input events and linguistic aid events from a plurality of text input device formats through the at least one application program and providing a plurality of text service event handlers in response to the received data.” Col. 1, ll. 57-65 (emphasis added).
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	“In one system aspect, the system includes at least one application program, and an operating system supporting operations of the at least one application program, the operating system comprising a text services manager for receiving data including text input events and linguistic aid events from a plurality of text input device formats through the at least one application program and providing a plurality of text service event handlers in response to the received data.” Col. 1, ll. 57-65 (emphasis added). “Preferably, the application program 14 is unaware of the text services receiving and handling the events.” Col. 4, ll. 46-48. “The handling of events as described with reference to FIGS. 4a and 4b preferably occurs transparently to the application program 14. However, a protocol suitably exists to allow communication between a particular text service and the application to facilitate event handling.” Col. 6, ll. 10-15.

Claim	U.S. Patent 7,411,582	
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	See Claims 19.1 and 19.2.

Claim	U.S. Patent 7,411,582	
21	21. The system of claim 19 further comprising an input panel window corresponding to the selected input method.	See Claims 1.3 and 1.4.

Claim	U.S. Patent 7,411,582	
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	See Claims 1.3 and 1.4.

Claim	U.S. Patent 7,411,582	
23	23. The system of claim 21 wherein the manager component selectively displays and hides the input panel window.	See Claim 14. “An application program 14 preferably interacts with the TSM 20 with the creation of a data structure (TSMDocument) i.e., an opaque structure in memory that organizes related pieces of information, the basic types of which include files, lists, arrays, records, trees, and tables, as is well understood by those skilled in the art. The application program 14 further preferably performs operations, such as activate and deactivate, on the data structure, where activate suitably refers to activation of an input method and insertion of a menu for the TSM 20 into the application program 14 menu bar, while deactivate suitably refers to deactivation of an input method and removal of the TSM's menu from the menu bar for the application program 14.” Col. 3, ll. 53-65.

Claim	U.S. Patent 7,411,582	
25	25. The system of claim 19 where the	“Processing of text inputs into a computer system has many facets. Application

Claim	U.S. Patent 7,411,582	programs typically handle keystroke entries from a keyboard, as well as selection of characters from a keyboard palette displayed on a display device through the use of a mouse or other pointing device. On-going technological improvements to computer systems have enhanced the methods for inputting data, including text, into computers. Such improvements include the ability to input text as handwriting, via a pen or stylus device. Another improvement includes the ability to input text captured as speech, via a microphone input device.” Col. 1, ll. 12-22.

Claim	U.S. Patent 7,411,582	“The handling of events as described with reference to FIGS. 4a and 4b preferably occurs transparently to the application program 14. However, a protocol suitably exists to allow communication between a particular text service and the application to facilitate event handling.” Col. 6, ll. 10-15.
26	26. The system of claim 19 wherein the manager component transfers information from the computer program to the selected input method.	

Claim	U.S. Patent 7,411,582	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system capable of calling functions in the manager component via a defined interface set. Col. 14, ll. 23-29. In addition, the combination of Saunders ‘499 and U.S. Patent 5,157,384 (“Greanias ‘384”) renders this claim obvious. Greanias ‘384 teaches calling functions in the manager component via a defined interface set. See Greanias ‘384 Claim 37 and Col. 8, ll. 59-65. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined interface set.	


Claim	U.S. Patent 7,411,582	
		Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.


Claim	U.S. Patent 7,411,582	
28	28. The system of claim 19 wherein the selected input method comprises an object.	<p>“As shown, upon the occurrence of an event to be handled by a text service, an application program 14 makes a call for a new data structure, e.g., TSMDocument, from the TSM 20 (step 30). In turn, the data structure TSMDocument is suitably created (step 32). Preferably, the data structure provides a high level interface to text services, and gives the application program 14 access to in-line processing of inputs.” Col 1, ll. 1-8.</p> <p>“A SOM object 44 for the text service is created by the TSM 20 when the text service is created and is pointed to by the second data structure TSMContext 40. The SOM object 44 suitably provides methods necessary for interaction between the TSM 20 and the text service to which it refers. Although described as a SOM object, i.e., a mechanism created in accordance with SOM, from for example, IBM Corporation, New York, it is also suitably created and implemented using other languages, including C, C++, and the like, as is well appreciated by those skilled in the art.” Col. 4, ll. 33-44.</p>

Claim	U.S. Patent 7,411,582	
29	29. The system of claim 19 wherein the selected input method draws an input panel in an input panel window displayed in the graphical windowing environment.	“The present invention provides system and method aspects to support a plurality of textual manipulations and inputs and provide appropriate textual outputs. In one system aspect, the system includes at least one application program, and an operating system supporting operations of the at least one application program, the operating

Claim	U.S. Patent 7,411,582		<p>system comprising a text services manager for receiving data including text input events and linguistic aid events from a plurality of text input device formats through the at least one application program and providing a plurality of text service event handlers in response to the received data.” Col. 1, ll. 55-65.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to draw an input panel in an input panel window. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders ‘499 and the MessagePad 120 renders this claim obvious. As shown below, the MessagePad 120 draws an input panel in an input panel window. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <div data-bbox="834 533 1328 926" data-label="Image"> </div> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified,</p>
-------	-----------------------	--	---

Claim	U.S. Patent 7,411,582	
		predictable solutions.

Claim	U.S. Patent 7,411,582	
30	30. The system of claim 29 wherein the manager component selectively displays and hides the display of the input panel window.	<p>See Claim 29.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to permit the manager component to selectively display and hide the display of the input panel window. For example, Saunders '499 discloses several input methods, and it would have been obvious to one of skill in the art to permit selection of a particular input method by the user. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders '499 and the MessagePad 120 renders this claim obvious. As shown below, when the user selects the Numeric input method option on the MessagePad 120, the Typewriter input method window closes, and the Numeric input method window opens.</p> 

Claim	U.S. Patent 7,411,582	
		 <p>One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	
31	31. The system of claim 29 wherein the manager component docks the input panel window.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 7, ll. 44-54.</p> <p>In addition, the combination of Saunders '499 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to</p>

Claim	U.S. Patent 7,411,582	<p>work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
-------	-----------------------	---

EXHIBIT M

Exemplar Chart of U.S. Patent 6,339,780**Apple HyperCard¹
Claims 1-6, 9-14, 17, 18, 20, 21 and 32-42**

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or U.S. Patent 6,023,698 (“Lavey ‘698’”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978’”) and/or U.S. Patent 5,864,850 (“Nordman ‘850’”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348’”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843’”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
1	A hypermedia browser embodied on a computer-readable medium for execution on an information processing device	HyperCard includes a hypermedia browser capable of loading additional or alternative hypermedia content in response to a user’s selection of hyperlinks.

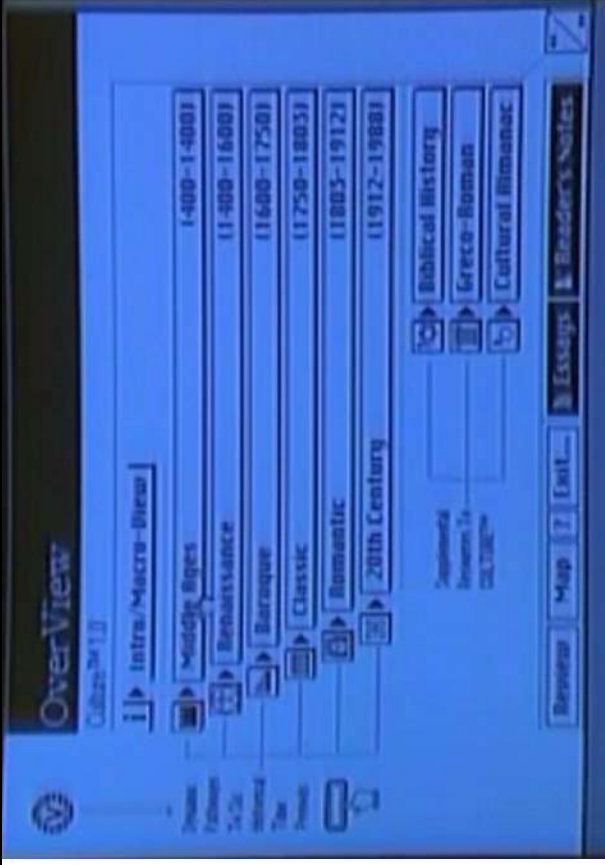
NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ Apple HyperCard; described, e.g., in (i) an episode of “The Computer Chronicles” (entitled “HyperCard Update”) which, on information and belief, was aired on KCSM TV in San Mateo, CA on August 1, 1990; (ii) Macintosh HyperCard User’s Guide, Apple Computer, Inc. 1988; and (iii) Macintosh HyperCard User’s Guide, Apple Computer, Inc. 1987.

² “Dynamic Progress Marking Icon” issued to Jaaskelainen; filed July 16, 1992; issued April 5, 1994.

³ “Replaceable and Extensible Navigator Component of a Network Component System” issued to Cleron et al.; filed February 27, 1997; issued May 25, 1999.

⁴ PenPoint Operating System, PenPoint UI Design Guidelines; GO Corporation, Foster City, California; 1991.

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	having a limited display area,	 <p style="text-align: right;">FIG. 1</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser on an information processing device having a limited display area.</p> <p>In addition, the combination of HyperCard and one or more references describing handheld devices with browsers renders this claim obvious. Gessler⁵, Cooper⁶,</p>

⁵ S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. Computer Networks and ISDN Systems 28 (1995) 53-59 (“Gessler”) (see pgs. 53 and 56-58).

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		Bartlett ⁷ , Watson ⁸ , Kamba ⁹ , and Lauff ¹⁰ each teach a hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area (e.g., a PDA, handheld, mobile device, etc.), and it would be obvious to one of skill in the art to modify HyperCard with these teachings. ¹¹ One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Additionally, Bartlett cites Gessler and Cooper cites Gessler and Bartlett. Moreover, all of these references are from the same general time period. Additionally, a person of ordinary skill in the art would

(...Continued)

⁶ I. Cooper and R. Shufflebotham. PDWeb Browsers: Implementation Issues. (University of Kent at Canterbury, November 9, 1995) (“Cooper”) (see pgs. 3 and 5).

⁷ J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE COMPCON 95 (San Francisco, March 1995) (“Bartlett”) (see pgs. 1, 4 and 5).

⁸ T. Watson. “Application Design for Wireless Computing” (1994 Mobile Computing Systems and Applications Workshop Position Paper (August 1994)).

⁹ T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). “Using small screen space more efficiently.”

¹⁰ M. Lauff and W. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.

¹¹ Motorola continues to investigate other browsers that were in existence prior to the priority date of the ‘780 patent (e.g., NetHopper, Newt’s Cape, PocketWeb and Vagon).

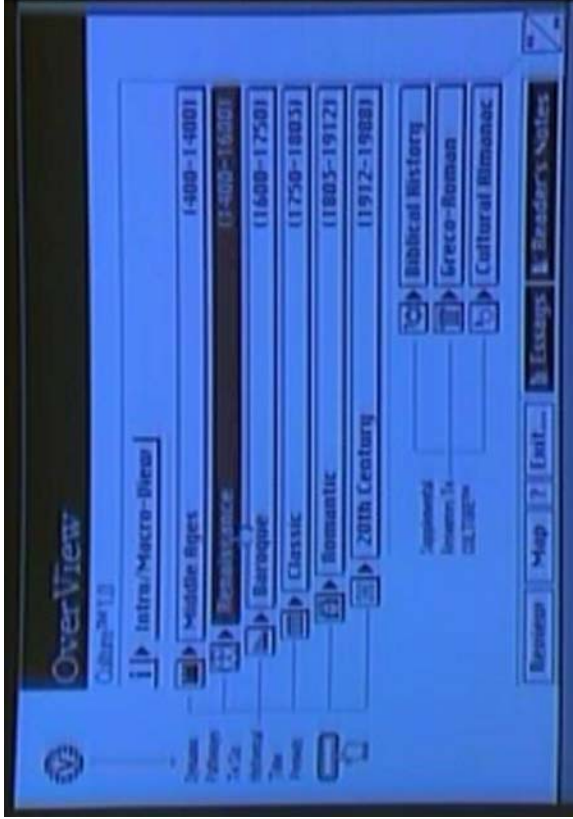
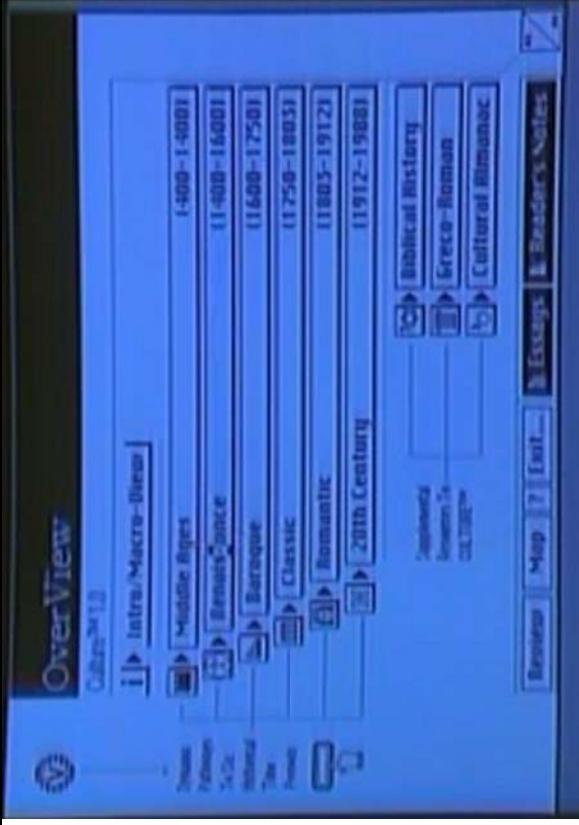

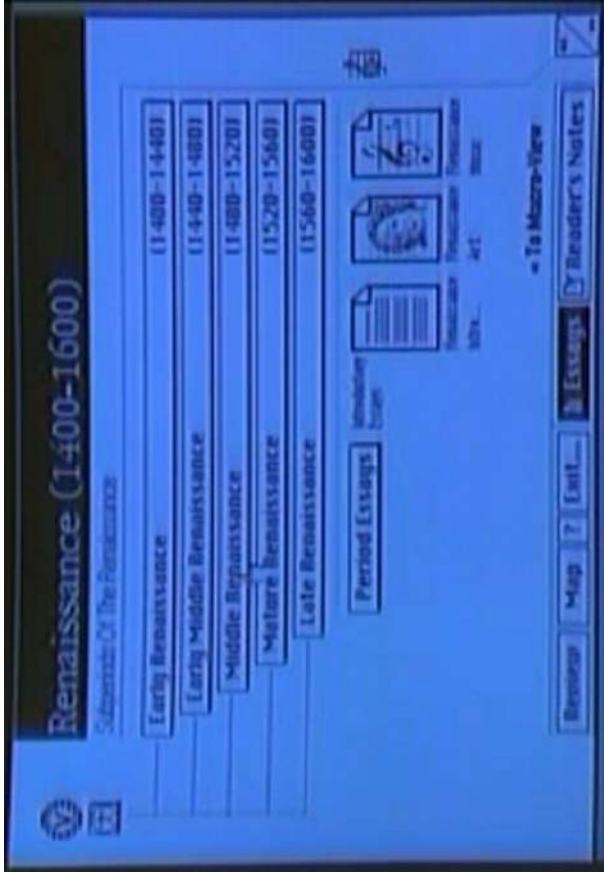
Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Claron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.1	wherein the hypermedia browser has a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading content,	HyperCard provides a content viewing area for viewing content. When one of the hyperlinks (e.g., Renaissance) is selected, the browser displays an icon (e.g., a watch) while the browser is loading content.  The image is a screenshot of a HyperCard stack titled 'Overview' with a subtitle 'Culture 101'. It features a hierarchical menu structure. The main menu includes 'Intro/Macro-View', 'Middle Ages', 'Renaissance', 'Baroque', 'Classic', 'Romantic', and '20th Century'. The 'Renaissance' menu is currently selected and highlighted in red, showing a sub-menu with '1400-1499', '1500-1599', '1600-1699', '1700-1799', '1750-1803', '1805-1912', and '1912-1988'. To the right of the main menu, there are buttons for 'Bible of History', 'French-Roman', and 'Cultural Renaissance'. At the bottom, there are buttons for 'Review', 'Map', 'List', 'Exit', 'Essays', and 'Reader's Notes'.

FIG. 2

Claim	U.S. Patent 6,339,780	<p>Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”)² and/or U.S. Patent 5,907,843 (“Cleron ‘843”)³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
		 <p style="text-align: right;">FIG. 3</p>

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) and/or U.S. Patent 5,907,843 (“Cleron ‘843”) and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		 <p style="text-align: right;">FIG. 4</p>
1.2	wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area,	The temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area. See the watch in FIGS. 3 and 4.
1.3	wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.	The temporary graphic element (e.g., the watch) is not content. The content displayed by HyperCard comes from a source external to the HyperCard browser (e.g., a stack). The FIGS. above are from Culture 1.0, an educational program that ran on HyperCard. Cards are HyperCard's basic unit of information, and stacks are collections of cards (HyperCard User's Guide at pp. 3-4). Stacks are often stored externally to the

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		HyperCard browser, often in a folder on a desktop or other storage medium (HyperCard User’s Guide at p. 160).
2	A hypermedia browser as recited in claim 1, wherein the browser is configured to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.	<p>The temporary graphic element (e.g., the watch) returns to a pointing-finger icon once the content is loaded.</p>  <p style="text-align: right;">FIG. 5</p>
3	A hypermedia browser as recited in claim 1, wherein the temporary graphic element indicates to a user that the browser is loading content.	As illustrated above, the temporary graphic element (e.g., the watch) is displayed while content is being loaded to indicate to a user that the browser is loading content.

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
4	A hypermedia browser as recited in claim 1, wherein the temporary graphic element disappears when the browser’s loading of content is complete to a user that such loading of content is complete.	See FIG. 5 above. The watch disappears when the browser’s loading of content is complete.
5	A hypermedia browser as recited in claim 1, wherein the temporary graphic element is animated.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to animate the temporary graphic element. In addition, the combination of HyperCard and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 and/or Jaaskelainen ‘348 and/or PenPoint renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 and/or Jaaskelainen ‘348 and/or PenPoint teaches a animating the temporary graphic element (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart; see also discussion of “dynamic progress marker icon” in Jaaskelainen ‘348 and discussion of “Animated Busy Clock” in PenPoint at pages 64-65) and it would be obvious to one of skill in the art to modify HyperCard with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
6	A hypermedia browser as recited in claim 1, wherein the hypermedia browser	The watch icon is the mouse cursor and could be positioned by the user in a corner of the content viewing area.

Claim	U.S. Patent 6,339,780	<p>Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”)² and/or U.S. Patent 5,907,843 (“Cleron ‘843”)³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
	displays the temporary graphic element in a corner of the content viewing area.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area. Displaying the temporary graphic element in a corner of the content viewing area is an obvious design choice to minimize the amount of content obstructed and would have been obvious to try.</p> <p>In addition, the combination of HyperCard and/or Vaughton ‘744 and/or Jaaskelainen ‘348 and/or Cleron ‘843 renders this claim obvious. Vaughton ‘744 and/or Jaaskelainen ‘348 and/or Cleron ‘843 teaches displaying the temporary graphic element in a corner of the content viewing area (see analysis of Vaughton ‘744 in claim chart; see also Jaaskelainen ‘348, which indicates, e.g., that “[i]n the preferred embodiment, icon 20 is displayed in the lower right hand corner or display 12 (FIG. 1), and remains in the same location for the duration of the task. Of course, icon 20 could be displayed in a different location of display 12 or move from location to location as the task progresses.” Jaaskelainen ‘348 at Col. 4, ll. 7-13; and Cleron ‘843, which indicates, e.g., that progress banner 750 may appear in the lower right corner of the display. Cleron ‘843 at FIG. 8A.) and it would be obvious to one of skill in the art to modify HyperCard with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
9	A hypermedia browser as recited in	When one of the hyperlinks (e.g., Renaissance) is selected, the browser displays an

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	claim 1, wherein the temporary graphic element conveys status information of the browser.	icon (e.g., a watch) that conveys status information of the browser, e.g., that the browser is busy retrieving the requested information. See, e.g., FIGS. 2-4 above.
10	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	HyperCard used its own scripting language known as HyperTalk (HyperCard User’s Guide at xxv).
11	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement HyperCard wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web. In addition, the combination of HyperCard and Judson ‘643 and/or Lavey ‘698 and/or Nguyen ‘498 and/or Nguyen ‘978 renders this claim obvious. Judson ‘643 and/or Lavey ‘698 and/or Nguyen ‘498 and/or Nguyen ‘978 and/or Nordman ‘850 teaches content that is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web (see analysis of Judson ‘643 and/or Lavey ‘698 and/or Nguyen ‘498 and/or Nguyen ‘978 and/or Nordman ‘850 in claim chart) and it would be obvious to one of skill in the art to modify HyperCard with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
12	An information processing device comprising:	HyperCard operated on Apple Macintosh computers. In 1988, HyperCard ran on the Macintosh Plus (System 3.2), Macintosh SE (System 4.0) and Macintosh II (System 4.1). These computer systems running HyperCard were information processing devices in accordance with the elements of claim 12.
12.1	a processor;	The Macintosh computer had a processor that was used to run HyperCard.
12.2	a display;	HyperCard’s output was displayed on the monitor display of the Macintosh computer.
12.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser has a content viewing area for viewing content and wherein the hypermedia browser displays a temporary graphic element over the content viewing area during times when the browser is loading visible content;	<i>See</i> claim 1.1.
12.4	wherein the temporary graphic element is positioned only over a portion of the content viewing area and obstructs only part of the visible content in the content	<i>See</i> claim 1.2.

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	viewing area; and	
12.5	wherein the temporary graphic element indicates to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser.	The temporary graphic element (<i>e.g.</i> , the watch) indicates to a user that the browser is loading content (<i>See</i> claim 1.1). HyperCard content comprises data for presentation which is from a source external to the browser (<i>See</i> claim 1.3).
13	An information processing device as recited in claim 12, wherein the temporary graphic element is animated.	<i>See</i> claim 5.
14	An information processing device as recited in claim 12, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	<i>See</i> claim 6.
17	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<i>See</i> claim 10.
18	A hypermedia browser of claim 12, wherein content is data formatted for	<i>See</i> claim 11.

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	
20	An information processing device as recited in claim 12, wherein the temporary graphic element is not content.	See claim 1.3.
21	An information processing device as recited in claim 12, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.	See claim 4.
32	A method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:	HyperCard provides a content viewing area for viewing content. See, e.g., FIG. 1 above. When one of the hyperlinks (e.g., Renaissance) is selected, the browser displays an icon (e.g., a watch) indicating a content “load status” of the hypermedia browser. See, e.g., FIGS. 2-4 above.
32.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a “load status” graphic element,	See FIGS. 2-4 above.

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;	
32.2	receiving an instruction to load new content into the content viewing area;	See FIG. 2 above which shows selecting “Renaissance” for loading.
32.3	loading such new content into the content viewing area; and	See FIGS. 3-5 above which show the Renaissance page being loaded.
32.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	See FIGS. 3 and 4 above which show a watch icon over the content viewing area, obstructing only part of the content in the content viewing area.
32.5	wherein content comprises data for presentation which is from a source external to the browser.	See claim 1.3.
33	A method as recited in claim 32 further comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	See FIGS. 3-4 above which show the watch disappearing upon completion of the loading.
34	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a	See claim 10.

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	markup language, visible text of such a markup language, and visible results of a scripting language.	
35	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.
36	A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:	HyperCard is a computer-readable medium having computer-executable instructions. HyperCard provides a content viewing area for viewing content. <i>See, e.g.</i> , FIG. 1 above. When one of the hyperlinks (<i>e.g.</i> , Renaissance) is selected, the browser displays an icon (<i>e.g.</i> , a watch) indicating a content “load status” of the hypermedia browser. <i>See, e.g.</i> , FIGS. 2-4 above.
36.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen is without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of	See claim 32.1.

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	the hypermedia browser;	
36.2	receiving an instruction to load new content into the content viewing area;	<i>See</i> claim 32.2.
36.3	loading such new content into the content viewing area; and	<i>See</i> claim 32.3.
36.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	<i>See</i> claim 32.4.
36.5	wherein content comprises data for presentation which is from a source external to the browser.	<i>See</i> claim 32.5.
37	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<i>See</i> claim 10.
38	A hypermedia browser of claim 36,	<i>See</i> claim 11.

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) and/or U.S. Patent 5,907,843 (“Cleron ‘843”) and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	
39	A computer-readable medium as recited in claim 36 further having additional computer-executable instructions that perform a method comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	See claim 33.
40	An information processing device comprising:	See claim 12.
40.1	a processor;	See claim 12.1.
40.2	a display;	See claim 12.2.
40.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display;	See FIGS. 1-5 above.

Claim	U.S. Patent 6,339,780	Apple HyperCard alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	wherein the hypermedia browser is configured to operate in a content-loading mode and a content-loaded mode;	
40.4	in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no “load status” graphic element is displayed, wherein absence of such “load status” graphic element indicates that the browser is in the content-loaded mode;	See FIGS. 1 and 5 above.
40.5	in the content-loading mode, the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a “load status” graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such “load status” graphic element indicates that the browser is in the content-loading mode; and	See FIGS. 3 and 4 above.
40.6	wherein content comprises data for presentation which is from a source external to the browser.	See claim 32.5.

EXHIBIT N

Exemplar Chart of U.S. Patent 6,339,780**U.S. Patent 5,845,282 (“Alley ‘282”)¹
Claims 1-6, 9-14, 17, 18, 20, 21 and 32-42**

Claim	U.S. Patent 6,339,780	Alley ‘282 alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or U.S. Patent 5,584,498 (“Nguyen ‘498”) and/or U.S. Patent 5,864,850 (“Nordman ‘850”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”) ² and/or U.S. Patent 5,907,843 (“Cleron ‘843”) ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
1	A hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area,	“A method and apparatus for selecting and retrieving computer data files from a remote computer includes an automatic file translation mechanism. In one embodiment, the data being retrieved and the file translation mechanism are located on the same computer. The method of the invention includes establishing a data transfer link with the remote computer, displaying the files available for retrieval from

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Method and Apparatus for Remotely Accessing Files From a Desktop Computer Using a Personal Digital Assistant” issued to Alley et al.; filed August 7, 1995; issued December 1, 1998.

² “Dynamic Progress Marking Icon” issued to Jaaskelainen; filed July 16, 1992; issued April 5, 1994.

³ “Replaceable and Extensible Navigator Component of a Network Component System” issued to Cleron et al.; filed February 27, 1997; issued May 25, 1999.

⁴ PenPoint Operating System, PenPoint UI Design Guidelines; GO Corporation, Foster City, California; 1991.

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
		<p>the remote computer, selecting a file to be transferred, and translating the file prior to transfer. In one embodiment, the apparatus includes a pen-based computer and the remote computer is a desktop computer." Abstract.</p> <p>"Even smaller hand-held computers are now capable of computing tasks that required much larger machines a few short years ago." col. 1, ll. 17-19.</p> <p>"In one aspect, the present invention provides a method for transferring data from a first computer system running a server program under a first operating system to a second computer system running a control program under a second operating system, the first and second computer systems being in communication through a communications medium. In one embodiment, a data transfer link from the control program on the second computer system to the server program on the first computer system is established. The second computer system then displays a list of files available on the first computer system which are available for translation and transfer. A file on the first computer system is then selected for transfer. The file is then translated and transferred to the second computer system." col. 2, ll. 30-36.</p> <p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		<p>include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser on an information processing device having a limited display area.</p> <p>In addition, the combination of Alley '282 and one or more references describing handheld devices with browsers renders this claim obvious. Gessler⁵, Cooper⁶, Bartlett⁷, Watson⁸, Kamba⁹, and Lauff¹⁰ each teach a hypermedia browser embodied on a computer-readable medium for execution on an information processing device</p>

⁵ S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. Computer Networks and ISDN Systems 28 (1995) 53-59 ("Gessler") (see pgs. 53 and 56-58).

⁶ I. Cooper and R. Shufflebotham. PDWeb Browsers: Implementation Issues. (University of Kent at Canterbury, November 9, 1995) ("Cooper") (see pgs. 3 and 5).

⁷ J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE COMPCON 95 (San Francisco, March 1995) ("Bartlett") (see pgs. 1, 4 and 5).

⁸ T. Watson. "Application Design for Wireless Computing" (1994 Mobile Computing Systems and Applications Workshop Position Paper (August 1994)).

⁹ T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). "Using small screen space more efficiently."

¹⁰ M. Lauff and W. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		having a limited display area (e.g., a PDA, handheld, mobile device, etc.), and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. ¹¹ One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Additionally, Bartlett cites Gessler and Cooper cites Gessler and Bartlett. Moreover, all of these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.1	wherein the hypermedia browser has a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading content,	<i>See</i> FIGS. 3, 4 and 10A-D. <p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"Another preferred tool for implementing the system of the present invention is a view system. Various types of view systems are well known to those skilled in the art. In the present system, the notepad application on the screen 60 can form a first or 'root' layer, with the status bar 160, for example, positioned in a second layer 'over' the root</p>

¹¹ Motorola continues to investigate other browsers that were in existence prior to the priority date of the '780 patent (e.g., NetHopper, Newt's Cape, PocketWeb and Vagon).

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
		<p>layer. The various buttons 162 of the status bar 160 are positioned in a third layer 'over' the second and root layers." col. 9, ll. 10-17.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p> <p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>"Should the user chose button 322, the sequence of actions described above with respect to FIGS. 9A and 9B leads to the display shown in FIG. 10D. As shown in the Figure, window 317 is overlaid by a new window 324 which window displays the desktop file information for the file "HeapShow.pkg" as shown generally at 326." col. 13, ll. 56-61.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a temporary graphic element over the content viewing area during times when the browser is loading content. For example, Alley discloses several status indicators above. Thus one of skill in the art would have been motivated to display a temporary graphic element over the content viewing area during times when the browser is loading content.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 renders this claim obvious. Vaughton '744 and/or Judson '643</p>

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p> <p>and/or Lavey '698 teaches displaying a temporary graphic element over the content viewing area during times information is being downloaded (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are each in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.2	wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area,	<p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>"Should the user chose button 322, the sequence of actions described above with respect to FIGS. 9A and 9B leads to the display shown in FIG. 10D. As shown in the Figure, window 317 is overlaid by a new window 324 which window displays the desktop file information for the file "HeapShow.pkg" as shown generally at 326." col. 13, ll. 56-61.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to position the temporary graphic element over the content viewing area to obstruct only part of the content in the content viewing area.</p> <p>In addition, the combination of Alley '282 and/or Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or PenPoint renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or PenPoint teaches a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area (see</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart; see also Jaaskelainen '348 at FIG. 1 and PenPoint at 64-65 (describing the placement of the animated busy clock)) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.3	wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.	<p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent</p>

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
		<p>to those of skill in the computer software arts." col. 10, ll. 24-34.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 teaches a temporary graphic element that is not content (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
2	<p>A hypermedia browser as recited in claim 1, wherein the browser is configured to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.</p>	<p>"Following the establishment of the connection, the list of files available from the desktop of the remote Macintosh is presented to the user in a dialog box 309 shown in FIG. 10B. This dialog box replaces dialog boxes 302 and 304 which are present only to describe to the user the status of the pending connection." col. 13, ll. 30-35.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or Cleron '843 renders this claim</p>

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p> <p>obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or Cleron '843 teaches displaying the temporary graphic element over the content viewing area only during times when the browser is loading visible content (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart; see also Jaaskelainen at Col. 4, ll. 31-34 ("Icon 20 shown in FIG. 2U is displayed to the user momentarily to indicate completion of the task, and then the border is removed, thereby causing icon 20 to disappear from display 12."; Cleron '843 at Col. 16, ll. 50-55 ("When a process has been completed and a page of information is being displayed, the status and progress banner 750, the process display 752 and the stop button 754 are eliminated from the display because they no longer provide any useful information.")) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a temporary graphic element indicating to a user that the browser is loading content.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 teaches a temporary graphic element indicating to a user that the</p>
3	A hypermedia browser as recited in claim 1, wherein the temporary graphic element indicates to a user that the browser is loading content.	

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p> <p>browser is loading content (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
4	<p>A hypermedia browser as recited in claim 1, wherein the temporary graphic element disappears when the browser's loading of content is complete to indicate to a user that such loading of content is complete.</p>	<p>"Following the establishment of the connection, the list of files available from the desktop of the remote Macintosh is presented to the user in a dialog box 309 shown in FIG. 10B. This dialog box replaces dialog boxes 302 and 304 which are present only to describe to the user the status of the pending connection." col. 13, ll. 30-35.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a temporary graphic element that disappears when the browser's loading of content is complete to indicate to a user that such loading of content is complete.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or Cleron '843 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or Cleron '843 teaches a temporary graphic element that disappears when the browser's loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart; see also Jaaskelainen at Col. 4, ll. 31-34 ("Icon 20 shown in FIG. 2U is displayed to the user momentarily to indicate completion of the task, and then the border is removed, thereby causing icon 20 to disappear from display 12."; Cleron '843 at Col. 16, ll. 50-55 ("When a process has been completed and a page of</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		information is being displayed, the status and progress banner 750, the process display 752 and the stop button 754 are eliminated from the display because they no longer provide any useful information.") and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
5	A hypermedia browser as recited in claim 1, wherein the temporary graphic element is animated.	<p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to animate the temporary graphic element.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or PenPoint renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or PenPoint teaches animating the temporary graphic element (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart; see also discussion of "dynamic progress marker icon" in Jaaskelainen '348 and discussion of "Animated Busy Clock" in PenPoint at pages 64-65) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to</p>

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p> <p>combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
6	<p>A hypermedia browser as recited in claim 1, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.</p>	<p>See FIGS. 3 and 4.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area. Displaying the temporary graphic element in a corner of the content viewing area is an obvious design choice to minimize the amount of content obstructed and would have been obvious to try.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Jaaskelainen '348 and/or Cleron '843 renders this claim obvious. Vaughton '744 and/or Jaaskelainen '348 and/or Cleron '843 teaches displaying the temporary graphic element in a corner of the content viewing area (see analysis of Vaughton '744 in claim chart; see also Jaaskelainen '348, which indicates, e.g., that "[i]n the preferred embodiment, icon 20 is displayed in the lower right hand corner or display 12 (FIG. 1), and remains in the same location for the duration of the task. Of course, icon 20 could be displayed in a different location of display 12 or move from location to location as the task progresses." Jaaskelainen '348 at Col. 4, ll. 7-13; and Cleron '843, which indicates, e.g., that progress banner 750 may appear in the lower right corner of the display. Cleron '843 at FIG. 8A.) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
9	A hypermedia browser as recited in claim 1, wherein the temporary graphic element conveys status information of the browser.	<p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convey status information of the browser with a temporary graphic element.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 teaches conveying status information of the browser with a temporary graphic element (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
10	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a	"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus. The file

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
	scripting language.	<p>browser is coupled to a file selector that allows the user to identify at least one file to be transferred from the remote computer system to the computer apparatus. A translator that determines whether the selected file is to be translated in conjunction with the transfer of the file to the computer apparatus is coupled to the selector." col. 2, l. 63 - col. 3, l. 10.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Alley '282 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language. Alley '282 discloses an apparatus for selecting and receiving data stored on a remote computer system. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Alley '282 in the context of a hypermedia browser.</p> <p>In addition, the combination of Alley '282 and Nguyen '498 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 renders this claim obvious. Nguyen '498 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language (see analysis of Nguyen '498 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 in claim chart) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
11	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus. The file browser is coupled to a file selector that allows the user to identify at least one file to be transferred from the remote computer system to the computer apparatus. A translator that determines whether the selected file is to be translated in conjunction with the transfer of the file to the computer apparatus is coupled to the selector." col. 2, l. 63 - col. 3, l. 10.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Alley '282 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web. Alley '282 discloses an apparatus for selecting and receiving data stored on a remote computer system. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Alley '282 in the context of a hypermedia browser.</p> <p>In addition, the combination of Alley '282 and Nguyen '498 and/or Judson '643</p>

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
		<p>and/or Lavey '698 and/or Nordman '850 renders this claim obvious. Nguyen '498 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web (see analysis of Nguyen '498 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 in claim chart) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
12	An information processing device comprising:	<p>"As shown in FIG. 1, a block diagram 10 of the electronics of a pen-based computer in accordance with the present invention includes a central processing unit (CPU) 12, a memory system 14, an input/output (I/O) dual function display system 16, a clock system 18, a power system 20, a sound system 22, a PCMCIA connector 24, and a serial I/O system 26. The various components and systems of the computer 10 are coupled together by an I/O controller 28 which serves as an interface between the CPU 12 and other components of the computer 10. More specifically, the I/O controller 28 is an application-specific integrated circuit (ASIC) designed to handle memory, peripherals, and I/O tasks, as well as housekeeping functions such as providing system clocks, controlling power usage, etc. The design, manufacture, and use of ASICs is well known to those skilled in the art. The pen-based computer 10 as illustrated is currently being manufactured and sold by Apple Computer, Inc. of Cupertino, Calif. as a Newton.RTM. 120 Personal Digital Assistant (PDA)." col. 4,</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
12.1	a processor;	<p>Il. 33-51.</p> <p>"As shown in FIG. 1, a block diagram 10 of the electronics of a pen-based computer in accordance with the present invention includes a central processing unit (CPU) 12." col. 4, Il. 33-35.</p> <p>"CPU 12 is preferably a commercially available, single chip microprocessor. While CPU 12 can be a complex instruction set computer (CISC) chip, it is preferable that CPU 12 be one of the commercially available, reduced instruction set computer (RISC) chips which are known to be of generally higher performance than CISC chips. In the present embodiment, the CPU 12 is preferably an ARM.RTM. 610 RISC chip operating at 20 megahertz and is available from a variety of sources including VLSI Technology, Inc. of San Jose, Calif. and Plessey Semiconductor of England. The present CPU 12 includes a 32 bit data (D) bus 30, a 32 bit address (A) bus 32, and an 8 bit control (C) bus 34." col. 4, Il. 52-63.</p>
12.2	a display;	<p>"The display system 16 serves as both an input device and an output device." col. 5, Il. 19-20.</p> <p>"The display system 16 further includes an ASIC 56, a dedicated SRAM 58, and an LCD screen 60. The ASIC 56 is an LCD controller coupled to the data (D) bus 30, the address (A) bus 32, and the memory control bus 42. The purpose of the ASIC 56 is to allow the CPU 12 to write to the screen as if it were a RAM sitting on the memory bus 42. The SRAM 58 is coupled to the ASIC 56 by a dedicated bus 62, and the screen 60 is coupled to the ASIC 56 by a dedicated bus 64. The ASIC 56 serves as a controller for the screen 60, and uses the SRAM 58 as a frame buffer to store images to be displayed on the screen 60. The LCD screen 60 is preferably a standard super-twist LCD matrix screen available from a number of sources including Seiko-Epson of Japan. The LCD screen preferably comprises a rectangular array of picture elements or "pixels", as is well known to those skilled in the art." col. 5, Il. 33-47.</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
12.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser has a content viewing area for viewing content and wherein the hypermedia browser displays a temporary graphic element over the content viewing area during visible times when the browser is loading visible content;	<i>See</i> claim 1.1.
12.4	wherein the temporary graphic element is positioned only over a portion of the content viewing area and obstructs only part of the visible content in the content viewing area; and	<i>See</i> claim 1.2.
12.5	wherein the temporary graphic element indicates to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser.	<p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can</p>

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p> <p>include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p> <p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the temporary graphic element indicates to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 teaches a temporary graphic element indicating to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 in claim chart) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
13	An information processing device as recited in claim 12, wherein the temporary graphic element is animated.	See claim 5.

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
14	An information processing device as recited in claim 12, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	See claim 6.
17	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	See claim 10.
18	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.
20	An information processing device as recited in claim 12, wherein the temporary graphic element is not content.	See claim 1.3.

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
21	An information processing device as recited in claim 12, wherein the temporary graphic element disappears when the browser's loading of content is complete to indicate to a user that such loading of content is complete.	<i>See claim 4.</i>
32	A method of indicating a content "load status" of a hypermedia browser having a content viewing area for viewing content, the method comprising:	<p><i>See FIGS. 3, 4 and 10A-D.</i></p> <p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p> <p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>"Should the user chose button 322, the sequence of actions described above with</p>

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
		<p>respect to FIGS. 9A and 9B leads to the display shown in FIG. 10D. As shown in the Figure, window 317 is overlaid by a new window 324 which window displays the desktop file information for the file "HeapShow.pkg" as shown generally at 326." col. 13, ll. 56-61.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Alley '282 in a hypermedia browser having a content viewing area for viewing content. Hypermedia browsers download files from another system, and it would have therefore been obvious to one of skill in the art to implement Alley '282 in the context of a hypermedia browser.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 teaches indicating a content "load status" of a hypermedia browser having a content viewing area for viewing content (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 in claim chart) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
32.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a "load status" graphic element, wherein a "load status" graphic element	<p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	indicates a current content load status of the hypermedia browser;	<p>computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p> <p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>"Following the establishment of the connection, the list of files available from the desktop of the remote Macintosh is presented to the user in a dialog box 309 shown in FIG. 10B. This dialog box replaces dialog boxes 302 and 304 which are present only to describe to the user the status of the pending connection." col. 13, ll. 30-35.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art for the "load status" graphic element to indicate a current content load status of the hypermedia browser.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 teaches a "load status" graphic element indicating a current content load status of the hypermedia browser (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 in claim chart) and it would be obvious to one of skill in the art to</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
32.2	receiving an instruction to load new content into the content viewing area;	<p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p>
32.3	loading such new content into the content viewing area; and	"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p> <p>computer system and displays the file information to a user of the apparatus." col. 2, ll. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p>
32.4	<p>while loading, displaying a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and</p>	<p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>"Following the establishment of the connection, the list of files available from the desktop of the remote Macintosh is presented to the user in a dialog box 309 shown in FIG. 10B. This dialog box replaces dialog boxes 302 and 304 which are present only to describe to the user the status of the pending connection." col. 13, ll. 30-35.</p> <p>"Should the user chose button 322, the sequence of actions described above with respect to FIGS. 9A and 9B leads to the display shown in FIG. 10D. As shown in the Figure, window 317 is overlaid by a new window 324 which window displays the desktop file information for the file "HeapShow.pkg" as shown generally at 326." col. 13, ll. 56-61.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area.</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or PenPoint renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or PenPoint teaches displaying a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart; see also Jaaskelainen '348 at FIG. 1 and PenPoint at 64-65 (describing the placement of the animated busy clock)) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
32.5	wherein content comprises data for presentation which is from a source external to the browser.	See claim 1.3.
33	A method as recited in claim 32 further comprising, upon completion of the loading, removing the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	"Following the establishment of the connection, the list of files available from the desktop of the remote Macintosh is presented to the user in a dialog box 309 shown in FIG. 10B. This dialog box replaces dialog boxes 302 and 304 which are present only to describe to the user the status of the pending connection." col. 13, ll. 30-35. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to upon completion of the loading, remove the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or Cleron '843 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or Cleron '843 teaches upon completion of the loading, removing the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart; see also Jaaskelainen at Col. 4, ll. 31-34 ("Icon 20 shown in FIG. 2U is displayed to the user momentarily to indicate completion of the task, and then the border is removed, thereby causing icon 20 to disappear from display 12."; Cleron '843 at Col. 16, ll. 50-55 ("When a process has been completed and a page of information is being displayed, the status and progress banner 750, the process display 752 and the stop button 754 are eliminated from the display because they no longer provide any useful information.")) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
34	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	See claim 10.

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
35	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.
36	A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method of indicating a content "load status" of a hypermedia browser having a content viewing area for viewing content, the method comprising:	<p>"As shown in FIG. 1, a block diagram 10 of the electronics of a pen-based computer in accordance with the present invention includes a central processing unit (CPU) 12, a memory system 14, an input/output (I/O) dual function display system 16, a clock system 18, a power system 20, a sound system 22, a PCMCIA connector 24, and a serial I/O system 26. The various components and systems of the computer 10 are coupled together by an I/O controller 28 which serves as an interface between the CPU 12 and other components of the computer 10. More specifically, the I/O controller 28 is an application-specific integrated circuit (ASIC) designed to handle memory, peripherals, and I/O tasks, as well as housekeeping functions such as providing system clocks, controlling power usage, etc. The design, manufacture, and use of ASICs is well known to those skilled in the art. The pen-based computer 10 as illustrated is currently being manufactured and sold by Apple Computer, Inc. of Cupertino, Calif. as a Newton.RTM. 120 Personal Digital Assistant (PDA)." col. 4, ll. 33-51.</p> <p>See FIGS. 3, 4 and 10A-D.</p> <p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention</p>

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
		<p>comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p> <p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>"Should the user chose button 322, the sequence of actions described above with respect to FIGS. 9A and 9B leads to the display shown in FIG. 10D. As shown in the Figure, window 317 is overlaid by a new window 324 which window displays the desktop file information for the file "HeapShow.pkg" as shown generally at 326." col. 13, ll. 56-61.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Alley '282 in a hypermedia browser having a content viewing area for viewing content. Hypermedia browsers download files from another system, and it would have therefore been obvious to one of skill in the art to implement Alley '282 in the context of a hypermedia browser.</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 teaches indicating a content "load status" of a hypermedia browser having a content viewing area for viewing content (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 in claim chart) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
36.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen is without a "load status" graphic element, wherein a "load status" graphic element indicates a current content load status of the hypermedia browser;	<i>See claim 32.1.</i>
36.2	receiving an instruction to load new content into the content viewing area;	<i>See claim 32.2.</i>
36.3	loading such new content into the content viewing area; and	<i>See claim 32.3.</i>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
36.4	while loading, displaying a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	See claim 32.4.
36.5	wherein content comprises data for presentation which is from a source external to the browser.	See claim 32.5.
37	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	See claim 10.
38	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.
39	A computer-readable medium as recited	See claim 33.

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
	in claim 36 further having additional computer-executable instructions that perform a method comprising, upon completion of the loading, removing the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	
40	An information processing device comprising:	<i>See</i> claim 12.
40.1	a processor;	<i>See</i> claim 12.1.
40.2	a display;	<i>See</i> claim 12.2.
40.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser is configured to operate in a content-loading mode and a content-loaded mode;	<p><i>See</i> FIGS. 3 and 4.</p> <p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		<p>function is performed on the selected file. The functions executed at step 190 can include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p> <p>To the extent not expressly or inherently disclosed, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Nordman '850 teach a hypermedia browser configured to operate in a content-loading mode and a content-loaded mode (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are each in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
40.4	in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no "load status" graphic element is displayed, wherein absence of such "load status" graphic element indicates that the browser is in the content-loaded mode;	<p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop</p>

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
		<p>computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p> <p>"Following the establishment of the connection, the list of files available from the desktop of the remote Macintosh is presented to the user in a dialog box 309 shown in FIG. 10B. This dialog box replaces dialog boxes 302 and 304 which are present only to describe to the user the status of the pending connection." col. 13, ll. 30-35.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser displays loaded content in the content viewing area and no "load status" graphic element is displayed, wherein absence of such "load status" graphic element indicates that the browser is in the content-loaded mode.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or Cleron '843 renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or Cleron '843 teach the hypermedia browser displaying loaded content in the content viewing area and no "load status" graphic element is displayed, wherein absence of such "load status" graphic element indicates that the browser is in the content-loaded mode (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart; see also Jaaskelainen at Col. 4, ll. 31-34 ("Icon 20 shown in FIG. 2U is displayed to the user momentarily to indicate completion of the task, and then the border is removed, thereby causing icon 20 to disappear from display 12."; Cleron '843 at Col. 16, ll. 50-55 ("When a process has been completed and a page of information is being displayed, the status and progress banner 750, the process display 752 and the stop button 754 are eliminated from the display because they no longer</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		provide any useful information.") and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40.5	in the content-loading mode, the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a "load status" graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such "load status" graphic element indicates that the browser is in the content-loading mode; and	<p>"In a third aspect, the present invention provides an apparatus for selecting and receiving data stored on a remote computer system. The apparatus of the invention comprises, in one embodiment, means for establishing a data transfer link with the remote computer system. The means for establishing a data link are coupled to a remote file browser that receives file information for data stored on the remote computer system and displays the file information to a user of the apparatus." col. 2, l. 63 - col. 3, l. 4.</p> <p>"At step 186, a browser function is invoked by the user of the remote pen-based computer to select one or more files for transfer and, optionally, translation, or to load a software package. At step 188 the file to be transferred or loaded from the desktop computer to the remote pen-based computer is selected and, at step 190, the desired function is performed on the selected file. The functions executed at step 190 can include, but are not limited to, file transfer, file translation, file loading, file deletion, file copying, file printing, and file moving. Other suitable functions will be apparent to those of skill in the computer software arts." col. 10, ll. 24-34.</p> <p>"The status of the connection is shown in a second dialog box 304 in which the progress of the connection establishment is indicated by a 'barber pole' 306." col. 13, ll. 25-27.</p> <p>"Should the user chose button 322, the sequence of actions described above with</p>

Claim	U.S. Patent 6,339,780	<p>Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or U.S. Patent 5,907,843 ("Cleron '843")³ and/or PenPoint⁴ and/or various systems implementing a web browser on a mobile device</p>
		<p>respect to FIGS. 9A and 9B leads to the display shown in FIG. 10D. As shown in the Figure, window 317 is overlaid by a new window 324 which window displays the desktop file information for the file "HeapShow.pkg" as shown generally at 326." col. 13, ll. 56-61.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a "load status" graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such "load status" graphic element indicates that the browser is in the content-loading mode.</p> <p>In addition, the combination of Alley '282 and Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or PenPoint renders this claim obvious. Vaughton '744 and/or Judson '643 and/or Lavey '698 and/or Jaaskelainen '348 and/or PenPoint teach the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a "load status" graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such "load status" graphic element indicates that the browser is in the content-loading mode (see analysis of Vaughton '744 and/or Judson '643 and/or Lavey '698 in claim chart; see also Jaaskelainen '348 at FIG. 1 and PenPoint at 64-65 (describing the placement of the animated busy clock)) and it would be obvious to one of skill in the art to modify Alley '282 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so,</p>

Claim	U.S. Patent 6,339,780	Alley '282 alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744) and/or U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 5,584,498 ("Nguyen '498") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or U.S. Patent 5,907,843 ("Cleron '843") ³ and/or PenPoint ⁴ and/or various systems implementing a web browser on a mobile device
		and there were a finite number of identified, predictable solutions.
40.6	wherein content comprises data for presentation which is from a source external to the browser.	See claim 32.5.
41	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	See claim 10.
42	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.

EXHIBIT O

Exemplar Chart of U.S. Patent 6,339,780

U.S. Patent 5,864,850 (“Nordman ‘850’”)¹
Claims 1-6, 9-14, 17, 18, 20, 21 and 32-42

Claim	U.S. Patent 6,339,780	Nordman ‘850 alone and/or in combination with U.S. Patent 5,907,843 (“Cleron ‘843’”) and/or U.S. Patent 5,528,744 (“Vaughton ‘744’”) and/or U.S. Patent 5,845,282 (“Alley ‘282’”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348’”) ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
1	A hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area,	<p>“The other popular information service on the Internet is the Web. Instead of providing a user with a hierarchical list-oriented view of information, the Web provides the user with a ‘linked-hypertext’ view. Metaphorically, the Web perceives the Internet as a vast book of pages, each of which may contain pictures, text, sound, movies or various other types of data in the form of documents. Web documents are written in HyperText Markup Language (HTML) and Web servers transfer HTML documents to each other through the HyperText Transfer Protocol (HTTP).” col. 2, l. 60 - col. 3, l. 2.</p> <p>“As an example, a user ‘traverses’ the Web by following hot items of a page displayed on a graphical Web browser. These hot items are hypertext links whose presence are</p>

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Asynchronous-Event Opening Component of a Network Component System” issued to Nordman; filed February 27, 1997; issued January 26, 1999.

² “Dynamic Progress Marking Icon” issued to Jaaskelainen; filed July 16, 1992; issued April 5, 1994.

³ PenPoint Operating System, PenPoint UI Design Guidelines; GO Corporation, Foster City, California; 1991.

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
		<p>indicated on the page by visual cues, e.g., underlined words, icons or buttons. When a user follows a link (usually by clicking on the cue with a mouse), the browser displays the target pointed to by the link which, in some cases, may be another HTML document." col. 3, ll. 13-20.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser on an information processing device having a limited display area.</p> <p>In addition, the combination of Baker '416 and one or more references describing handheld devices with browsers renders this claim obvious. Gessler⁴, Cooper⁵, Bartlett⁶, Watson⁷, Kamba⁸, and Lauff⁹ each teach a hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area (e.g., a PDA, handheld, mobile device, etc.), and it</p>

⁴ S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. Computer Networks and ISDN Systems 28 (1995) 53-59 ("Gessler") (see pgs. 53 and 56-58).

⁵ I. Cooper and R. Shufflebotham. PDWeb Browsers: Implementation Issues. (University of Kent at Canterbury, November 9, 1995) ("Cooper") (see pgs. 3 and 5).

⁶ J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE COMPCON 95 (San Francisco, March 1995) ("Bartlett") (see pgs. 1, 4 and 5).

⁷ T. Watson. "Application Design for Wireless Computing" (1994 Mobile Computing Systems and Applications Workshop Position Paper (August 1994)).

⁸ T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). "Using small screen space more efficiently."

⁹ M. Lauff and W. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.

Claim	U.S. Patent 6,339,780	<p>Nordman ‘850 alone and/or in combination with U.S. Patent 5,907,843 (“Cleron ‘843”) and/or U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,845,282 (“Alley ‘282”) and/or U.S. Patent 5,301,348 (“Jaaskelainen ‘348”)² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p> <p>would be obvious to one of skill in the art to modify Baker ‘416 with these teachings.¹⁰ One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Additionally, Bartlett cites Gessler and Cooper cites Gessler and Bartlett. Moreover, all of these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.1	wherein the hypermedia browser has a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading content,	<p>“A conventional display monitor 232 having a display screen 235 is also connected to I/O circuitry 218 via cable 238. A pointer (cursor) 240 is displayed on windows 244 of the screen 235 and its position is controllable via the mouse 230 or the keyboard 224, as is well-known.” col. 8, ll. 31-36.</p> <p>“In accordance with the present invention, a novel network-oriented component layer 450 contains the underlying technology for implementing the extensible and replaceable network component system having an opening part component 475 that acts as a placeholder and displays progress during an asynchronous-event when navigating computer networks, such as the Internet.” col. 9, ll. 27-33.</p> <p>“Referring again to FIG. 6, the window object 630 and the graphic interface object 650 are elements of a graphical user interface of a network component system having a customizable framework for greatly enhancing the ability of a user to monitor and display progress and status information during the opening process over a computer network.” col. 14, ll. 53-58.</p>

¹⁰ Motorola continues to investigate other browsers that were in existence prior to the priority date of the ‘780 patent (e.g., NetHopper, Newt’s Cape, PocketWeb and Vagon).

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p> <p>"A CyberStream 710, which is a subclass that encapsulates the protocol knowledge required to download data referenced by objects of its companion CyberItem subclass, is used to provide progress information to a CyberDisplay part 720 and polls the CyberDisplay part 720 for progress information. For example, the progress information may include process mode data, process status data, and data on the percentage of the process that has been completed. A progress broadcaster object 730 and a progress receiver object 740 work in conjunction with a progress part 750 to monitor and display the progress information to a user. For instance, the progress broadcaster object 730 represents an asynchronous process, such as a download operation, and monitors the progress of that operation towards completion." col. 15, l. 55 - col. 16, l. 2.</p> <p>"Typical examples of monitored work include the functions necessary to complete downloading and e-mailing operations. For instance, in downloading operations, the progress broadcaster object 730 may monitor the number of bytes that have been downloaded to a disk and the total number of bytes that are expected to be downloaded." col. 16, ll. 8-14.</p> <p>"Referring to FIG. 8 for example, the progress part 750 may display progress information as a bar graph 752 and/or a numerical display 754. The progress part 750 obtains the status and progress information from its associated progress receiver 740 for displaying this information on the GUI to the user." col. 16, ll. 26-31.</p> <p>"As indicated in step S150, a progress broadcaster is attached to the transient opener part for displaying progress and status information for the referenced data during this Resolve method. This transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." col. 16, ll. 54-60.</p> <p>FIG. 8 displays a progress bar chart.</p> <p>"As indicated in step S150, a progress broadcaster is attached to the transient opener part for displaying progress and status information for the referenced data during this</p>
1.2	wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area,	

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p>
		<p>Resolve method. This transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." col. 16, ll. 54-60.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to position the temporary graphic element over the content viewing area to obstruct only part of the content in the content viewing area. Nordman '850 discloses that "[t]his transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." e.g., a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area.</p> <p>In addition, the combination of Nordman '850 and/or Vaughton '744 and/or Alley '282 and/or Jaaskelainen '348 and/or PenPoint renders this claim obvious. Vaughton '744 and/or Alley '282 and/or Jaaskelainen '348 and/or PenPoint teaches a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area (see analysis of Vaughton '744 and/or Alley '282 in claim chart; see also Jaaskelainen '348 at FIG. 1 and PenPoint at 64-65 (describing the placement of the animated busy clock)) and it would be obvious to one of skill in the art to modify Nordman '850 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
1.3	wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.	<p>"The other popular information service on the Internet is the Web. Instead of providing a user with a hierarchical list-oriented view of information, the Web provides the user with a 'linked-hypertext' view. Metaphorically, the Web perceives the Internet as a vast book of pages, each of which may contain pictures, text, sound, movies or various other types of data in the form of documents. Web documents are written in HyperText Markup Language (HTML) and Web servers transfer HTML documents to each other through the HyperText Transfer Protocol (HTTP)." col. 2, l. 60 - col. 3, l. 2.</p> <p>"Referring to FIG. 8 for example, the progress part 750 may display progress information as a bar graph 752 and/or a numerical display 754. The progress part 750 obtains the status and progress information from its associated progress receiver 740 for displaying this information on the GUI to the user." col. 16, ll. 26-31.</p>
2	A hypermedia browser as recited in claim 1, wherein the browser is configured to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.	<p>See FIG. 9.</p> <p>"An important function of the opening process for the invention is to display progress to the user during an asynchronous process and to create a display part to display data when it is available. FIG. 9 is a flow chart illustrating this function." col. 16, ll. 32-35.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.</p> <p>In addition, Nordman '850 identifies U.S. patent application Ser. No. 08/806,483 (now Cleron et al. U.S. Patent No. 5,907,843 ("Cleron '843")), also filed February 27, 1997, as a related copending U.S. application. col. 1, ll. 32-36. Cleron '843 states the following:</p> <p>"FIG. 8(B) illustrates an example of the navigator display part after a process has been completed. When a process has been completed and a page of information is being displayed, the status and progress banner 750, the process display 752 and the stop button 754 are eliminated from the display because they no longer provide any useful information. However, once another process is initiated, these displays and functions</p>

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device will be provided." col. 16, ll. 49-56.</p> <p>Thus, the combination of Nordman '850 and Cleron '843 renders this claim obvious. Nordman '850 and Cleron '843 teach displaying the temporary graphic element over the content viewing area only during times when the browser is loading visible content, and it would be obvious to one of skill in the art to combine these references. One of skill in the art would be motivated to combine these references because they are in the same field and share common subject matter, address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>"For instance, the progress broadcaster object 730 represents an asynchronous process, such as a download operation, and monitors the progress of that operation towards completion." col. 15, l. 66 - col. 16, l. 2.</p>
3	A hypermedia browser as recited in claim 1, wherein the temporary graphic element indicates to a user that the browser is loading content.	<p>See FIG. 9.</p> <p>"An important function of the opening process for the invention is to display progress to the user during an asynchronous process and to create a display part to display data when it is available. FIG. 9 is a flow chart illustrating this function." col. 16, ll. 32-35.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to have the temporary graphic element disappear when the browser's loading of content is complete to indicate to a user that such loading of content is complete.</p> <p>In addition, Nordman '850 identifies U.S. patent application Ser. No. 08/806,483 (now Cleron et al. U.S. Patent No. 5,907,843 ("Cleron '843")), also filed February 27, 1997, as a related copending U.S. application. col. 1, ll. 32-36. Cleron '843 states the following:</p>
4	A hypermedia browser as recited in claim 1, wherein the temporary graphic element disappears when the browser's loading of content is complete to indicate to a user that such loading of content is complete.	

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p> <p>"FIG. 8(B) illustrates an example of the navigator display part after a process has been completed. When a process has been completed and a page of information is being displayed, the status and progress banner 750, the process display 752 and the stop button 754 are eliminated from the display because they no longer provide any useful information. However, once another process is initiated, these displays and functions will be provided." col. 16, ll. 49-56.</p> <p>Thus, the combination of Nordman '850 and Cleron '843 renders this claim obvious. Nordman '850 and Cleron '843 teach having the temporary graphic element disappear when the browser's loading of content is complete to indicate to a user that such loading of content is complete, and it would be obvious to one of skill in the art to combine these references. One of skill in the art would be motivated to combine these references because they are in the same field and share common subject matter, address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
5	A hypermedia browser as recited in claim 1, wherein the temporary graphic element is animated.	<p>"Referring to FIG. 8 for example, the progress part 750 may display progress information as a bar graph 752 and/or a numerical display 754. The progress part 750 obtains the status and progress information from its associated progress receiver 740 for displaying this information on the GUI to the user." col. 16, ll. 26-31.</p>
6	A hypermedia browser as recited in claim 1, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display the temporary graphic element in a corner of the content viewing area.</p> <p>In addition, Nordman '850 identifies U.S. patent application Ser. No. 08/806,483 (now Cleron et al. U.S. Patent No. 5,907,843 ("Cleron '843")), also filed February 27, 1997, as a related copending U.S. application. col. 1, ll. 32-36. Cleron '843 states the following:</p>

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p> <p>"The navigator display part 700 additionally includes a status and progress banner 750, a process display 752 and a stop button 754. If the size of the CyberItem is not known, a "barber pole" type of display will be provided by the status and progress banner 750 that spins to show that progress is being made with respect to the current process. However, if the size of the CyberItem is known, a thermometer-type display will be provided by the status and progress banner 750 indicating more accurately the status and progress of the current process towards completion. The process display 752 provides a display indicating the process being performed." col. 16, ll. 35-46. <i>See also</i> FIG. 8A, which shows progress banner 750 in the corner of the display.</p> <p>Thus, the combination of Nordman '850 and Cleron '843 renders this claim obvious. Nordman '850 and Cleron '843 teach displaying the temporary graphic element in a corner of the content viewing area, and it would be obvious to one of skill in the art to combine these references. One of skill in the art would be motivated to combine these references because they are in the same field and share common subject matter, address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
9	A hypermedia browser as recited in claim 1, wherein the temporary graphic element conveys status information of the browser.	<p>"Referring to FIG. 8 for example, the progress part 750 may display progress information as a bar graph 752 and/or a numerical display 754. The progress part 750 obtains the status and progress information from its associated progress receiver 740 for displaying this information on the GUI to the user." col. 16, ll. 26-31.</p>
10	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a	<p>"Web documents are written in HyperText Markup Language (HTML) and Web servers transfer HTML documents to each other through the HyperText Transfer Protocol (HTTP)." col. 2, l. 66 - col. 3, l. 2.</p>

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
	markup language, and visible results of a scripting language.	
11	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	"Web documents are written in HyperText Markup Language (HTML) and Web servers transfer HTML documents to each other through the HyperText Transfer Protocol (HTTP)." col. 2, l. 66 - col. 3, l. 2.
12	An information processing device comprising:	"FIG. 2 illustrates a typical hardware configuration of a client 200 comprising a central processing unit (CPU) 210 coupled between a memory 214 and input/output (I/O) circuitry 218 by bidirectional buses 212 and 216. The memory 214 typically comprises random access memory (RAM) for temporary storage of information and read only memory (ROM) for permanent storage of the computer's configuration and 25 basic operating commands, such as portions of an operating system (not shown). As described further herein, the operating system controls the operations of the CPU 210 and client computer 200." col. 8, ll. 14-24.
12.1	a processor;	"FIG. 2 illustrates a typical hardware configuration of a client 200 comprising a central processing unit (CPU) 210 coupled between a memory 214 and input/output (I/O) circuitry 218 by bidirectional buses 212 and 216." col. 8, ll. 14-15.
12.2	a display;	"A conventional display monitor 232 having a display screen 235 is also connected to I/O circuitry 218 via cable 238." col. 8, ll. 31-33.
12.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser has a	See claim 1.1.

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
	content viewing area for viewing content and wherein the hypermedia browser displays a temporary graphic element over the content viewing area during times when the browser is loading visible content;	
12.4	wherein the temporary graphic element is positioned only over a portion of the content viewing area and obstructs only part of the visible content in the content viewing area; and	See claim 1.2.
12.5	wherein the temporary graphic element indicates to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser.	<p>"For instance, the progress broadcaster object 730 represents an asynchronous process, such as a download operation, and monitors the progress of that operation towards completion." col. 15, l. 66 - col. 16, l. 2.</p> <p>"The other popular information service on the Internet is the Web. Instead of providing a user with a hierarchical list-oriented view of information, the Web provides the user with a 'linked-hypertext' view. Metaphorically, the Web perceives the Internet as a vast book of pages, each of which may contain pictures, text, sound, movies or various other types of data in the form of documents. Web documents are written in HyperText Markup Language (HTML) and Web servers transfer HTML documents to each other through the HyperText Transfer Protocol (HTTP)." col. 2, l. 60 - col. 3, l. 2.</p>
13	An information processing device as recited in claim 12, wherein the temporary graphic element is animated.	See claim 5.
14	An information processing device as recited in claim 12, wherein the	See claim 6.

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
	hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	
17	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	See claim 10.
18	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.
20	An information processing device as recited in claim 12, wherein the temporary graphic element is not content.	See claim 1.3.
21	An information processing device as recited in claim 12, wherein the temporary graphic element disappears when the browser's loading of content is complete to indicate to a user that such	See claim 4.

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
	loading of content is complete.	
32	A method of indicating a content "load status" of a hypermedia browser having a content viewing area for viewing content, the method comprising:	<p>"The other popular information service on the Internet is the Web. Instead of providing a user with a hierarchical list-oriented view of information, the Web provides the user with a 'linked-hypertext' view. Metaphorically, the Web perceives the Internet as a vast book of pages, each of which may contain pictures, text, sound, movies or various other types of data in the form of documents. Web documents are written in HyperText Markup Language (HTML) and Web servers transfer HTML documents to each other through the HyperText Transfer Protocol (HTTP)." col. 2, l. 60 - col. 3, l. 2.</p> <p>"As an example, a user 'traverses' the Web by following hot items of a page displayed on a graphical Web browser. These hot items are hypertext links whose presence are indicated on the page by visual cues, e.g., underlined words, icons or buttons. When a user follows a link (usually by clicking on the cue with a mouse), the browser displays the target pointed to by the link which, in some cases, may be another HTML document." col. 3, ll. 13-20.</p> <p>"A conventional display monitor 232 having a display screen 235 is also connected to I/O circuitry 218 via cable 238. A pointer (cursor) 240 is displayed on windows 244 of the screen 235 and its position is controllable via the mouse 230 or the keyboard 224, as is well-known." col. 8, ll. 31-36.</p> <p>"In accordance with the present invention, a novel network-oriented component layer 450 contains the underlying technology for implementing the extensible and replaceable network component system having an opening part component 475 that acts as a placeholder and displays progress during an asynchronous-event when navigating computer networks, such as the Internet." col. 9, ll. 27-33.</p> <p>"Referring again to FIG. 6, the window object 630 and the graphic interface object 650 are elements of a graphical user interface of a network component system having a customizable framework for greatly enhancing the ability of a user to monitor and display progress and status information during the opening process over a computer</p>

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device network." col. 14, ll. 53-58.</p> <p>"A CyberStream 710, which is a subclass that encapsulates the protocol knowledge required to download data referenced by objects of its companion CyberItem subclass, is used to provide progress information to a CyberDisplay part 720 and polls the CyberDisplay part 720 for progress information. For example, the progress information may include process mode data, process status data, and data on the percentage of the process that has been completed. A progress broadcaster object 730 and a progress receiver object 740 work in conjunction with a progress part 750 to monitor and display the progress information to a user. For instance, the progress broadcaster object 730 represents an asynchronous process, such as a download operation, and monitors the progress of that operation towards completion." col. 15, l. 55 - col. 16, l. 2.</p> <p>"Typical examples of monitored work include the functions necessary to complete downloading and e-mailing operations. For instance, in downloading operations, the progress broadcaster object 730 may monitor the number of bytes that have been downloaded to a disk and the total number of bytes that are expected to be downloaded." col. 16, ll. 8-14.</p> <p>"Referring to FIG. 8 for example, the progress part 750 may display progress information as a bar graph 752 and/or a numerical display 754. The progress part 750 obtains the status and progress information from its associated progress receiver 740 for displaying this information on the GUI to the user." col. 16, ll. 26-31.</p> <p>"As indicated in step S150, a progress broadcaster is attached to the transient opener part for displaying progress and status information for the referenced data during this Resolve method. This transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." col. 16, ll. 54-60.</p>
32.1	displaying loaded content within the content viewing area of a screen of a	See FIG. 8 and the discussion that it may open in its own window: "As indicated in step S150, a progress broadcaster is attached to the transient opener part for displaying

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p>
	<p>hypermedia browser, the screen being without a "load status" graphic element, wherein a "load status" graphic element indicates a current content load status of the hypermedia browser;</p>	<p>progress and status information for the referenced data during this Resolve method. This transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." col. 16, ll. 54-60.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a "load status" graphic element, wherein a "load status" graphic element indicates a current content load status of the hypermedia browser.</p> <p>In addition, Nordman '850 identifies U.S. patent application Ser. No. 08/806,483 (now Cleron et al. U.S. Patent No. 5,907,843 ("Cleron '843")), also filed February 27, 1997, as a related copending U.S. application. col. 1, ll. 32-36. Cleron '843 states the following:</p> <p>"FIG. 8(B) illustrates an example of the navigator display part after a process has been completed. When a process has been completed and a page of information is being displayed, the status and progress banner 750, the process display 752 and the stop button 754 are eliminated from the display because they no longer provide any useful information. However, once another process is initiated, these displays and functions will be provided." col. 16, ll. 49-56.</p> <p>Thus, the combination of Nordman '850 and Cleron '843 renders this claim obvious. Nordman '850 and Cleron '843 teach displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a "load status" graphic element, wherein a "load status" graphic element indicates a current content load status of the hypermedia browser, and it would be obvious to one of skill in the art to combine these references. One of skill in the art would be motivated to combine these references because they are in the same field and share common subject matter, address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need</p>

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
		and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
32.2	receiving an instruction to load new content into the content viewing area;	"As an example, a user 'traverses' the Web by following hot items of a page displayed on a graphical Web browser. These hot items are hypertext links whose presence are indicated on the page by visual cues, e.g., underlined words, icons or buttons. When a user follows a link (usually by clicking on the cue with a mouse), the browser displays the target pointed to by the link which, in some cases, may be another HTML document." col. 3, ll. 13-20.
32.3	loading such new content into the content viewing area; and	"As an example, a user 'traverses' the Web by following hot items of a page displayed on a graphical Web browser. These hot items are hypertext links whose presence are indicated on the page by visual cues, e.g., underlined words, icons or buttons. When a user follows a link (usually by clicking on the cue with a mouse), the browser displays the target pointed to by the link which, in some cases, may be another HTML document." col. 3, ll. 13-20.
32.4	while loading, displaying a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	FIG. 8 displays a progress bar chart. <p>"As indicated in step S150, a progress broadcaster is attached to the transient opener part for displaying progress and status information for the referenced data during this Resolve method. This transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." col. 16, ll. 54-60.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area. Nordman '850 discloses that "[t]his transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." e.g., a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area. Based on this, one of skill in the art would have been</p>

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p> <p>motivated to implement a browser displaying a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area.</p> <p>In addition, the combination of Nordman '850 and/or Vaughton '744 and/or Alley '282 and/or Jaaskelainen '348 and/or PenPoint renders this claim obvious. Vaughton '744 and/or Alley '282 and/or Jaaskelainen '348 and/or PenPoint teaches displaying a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area (see analysis of Vaughton '744 and/or Alley '282 in claim chart; see also Jaaskelainen '348 at FIG. 1 and PenPoint at 64-65 (describing the placement of the animated busy clock)) and it would be obvious to one of skill in the art to modify Nordman '850 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>See claim 1.3.</p>
32.5	wherein content comprises data for presentation which is from a source external to the browser.	See FIG. 9.
33	A method as recited in claim 32 further comprising, upon completion of the loading, removing the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the	<p>"An important function of the opening process for the invention is to display progress to the user during an asynchronous process and to create a display part to display data when it is available. FIG. 9 is a flow chart illustrating this function." col. 16, ll. 32-35.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one</p>

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p> <p>of skill in the art to, upon completion of the loading, removing the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.</p> <p>In addition, Nordman '850 identifies U.S. patent application Ser. No. 08/806,483 (now Cleron et al. U.S. Patent No. 5,907,843 ("Cleron '843")), also filed February 27, 1997, as a related copending U.S. application. col. 1, ll. 32-36. Cleron '843 states the following:</p> <p>"FIG. 8(B) illustrates an example of the navigator display part after a process has been completed. When a process has been completed and a page of information is being displayed, the status and progress banner 750, the process display 752 and the stop button 754 are eliminated from the display because they no longer provide any useful information. However, once another process is initiated, these displays and functions will be provided." col. 16, ll. 49-56.</p> <p>Thus, the combination of Nordman '850 and Cleron '843 renders this claim obvious. Nordman '850 and Cleron '843 teach, upon completion of the loading, removing the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed, and it would be obvious to one of skill in the art to combine these references. One of skill in the art would be motivated to combine these references because they are in the same field and share common subject matter, address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
34	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a	See claim 10.

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
	markup language, and visible results of a scripting language.	
35	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.
36	A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method of indicating a content "load status" of a hypermedia browser having a content viewing area for viewing content, the method comprising:	<p>"FIG. 2 illustrates a typical hardware configuration of a client 200 comprising a central processing unit (CPU) 210 coupled between a memory 214 and input/output (I/O) circuitry 218 by bidirectional buses 212 and 216." col. 8, ll. 14-15.</p> <p>"The other popular information service on the Internet is the Web. Instead of providing a user with a hierarchical list-oriented view of information, the Web provides the user with a 'linked-hypertext' view. Metaphorically, the Web perceives the Internet as a vast book of pages, each of which may contain pictures, text, sound, movies or various other types of data in the form of documents. Web documents are written in HyperText Markup Language (HTML) and Web servers transfer HTML documents to each other through the HyperText Transfer Protocol (HTTP)." col. 2, l. 60 - col. 3, l. 2.</p> <p>"As an example, a user 'traverses' the Web by following hot items of a page displayed on a graphical Web browser. These hot items are hypertext links whose presence are indicated on the page by visual cues, e.g., underlined words, icons or buttons. When a user follows a link (usually by clicking on the cue with a mouse), the browser displays the target pointed to by the link which, in some cases, may be another HTML document." col. 3, ll. 13-20.</p>

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p>
		<p>"A conventional display monitor 232 having a display screen 235 is also connected to I/O circuitry 218 via cable 238. A pointer (cursor) 240 is displayed on windows 244 of the screen 235 and its position is controllable via the mouse 230 or the keyboard 224, as is well-known." col. 8, ll. 31-36.</p> <p>"In accordance with the present invention, a novel network-oriented component layer 450 contains the underlying technology for implementing the extensible and replaceable network component system having an opening part component 475 that acts as a placeholder and displays progress during an asynchronous-event when navigating computer networks, such as the Internet." col. 9, ll. 27-33.</p> <p>"Referring again to FIG. 6, the window object 630 and the graphic interface object 650 are elements of a graphical user interface of a network component system having a customizable framework for greatly enhancing the ability of a user to monitor and display progress and status information during the opening process over a computer network." col. 14, ll. 53-58.</p> <p>"A CyberStream 710, which is a subclass that encapsulates the protocol knowledge required to download data referenced by objects of its companion CyberItem subclass, is used to provide progress information to a CyberDisplay part 720 and polls the CyberDisplay part 720 for progress information. For example, the progress information may include process mode data, process status data, and data on the percentage of the process that has been completed. A progress broadcaster object 730 and a progress receiver object 740 work in conjunction with a progress part 750 to monitor and display the progress information to a user. For instance, the progress broadcaster object 730 represents an asynchronous process, such as a download operation, and monitors the progress of that operation towards completion." col. 15, l. 55 - col. 16, l. 2.</p> <p>"Typical examples of monitored work include the functions necessary to complete downloading and e-mailing operations. For instance, in downloading operations, the progress broadcaster object 730 may monitor the number of bytes that have been downloaded to a disk and the total number of bytes that are expected to be</p>

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device and/or various systems implementing a web browser on a mobile device downloaded." col. 16, ll. 8-14.
		<p>"Referring to FIG. 8 for example, the progress part 750 may display progress information as a bar graph 752 and/or a numerical display 754. The progress part 750 obtains the status and progress information from its associated progress receiver 740 for displaying this information on the GUI to the user." col. 16, ll. 26-31.</p> <p>"As indicated in step S150, a progress broadcaster is attached to the transient opener part for displaying progress and status information for the referenced data during this Resolve method. This transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." col. 16, ll. 54-60.</p>
36.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen is without a "load status" graphic element, wherein a "load status" graphic element indicates a current content load status of the hypermedia browser;	See claim 32.1.
36.2	receiving an instruction to load new content into the content viewing area;	See claim 32.2.
36.3	loading such new content into the content viewing area; and	See claim 32.3.
36.4	while loading, displaying a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	See claim 32.4.

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
36.5	wherein content comprises data for presentation which is from a source external to the browser.	See claim 32.5.
37	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	See claim 10.
38	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.
39	A computer-readable medium as recited in claim 36 further having additional computer-executable instructions that perform a method comprising, upon completion of the loading, removing the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element	See claim 33.

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
	obstructed when the element was displayed.	
40	An information processing device comprising:	See claim 12.
40.1	a processor;	See claim 12.1.
40.2	a display;	See claim 12.2.
40.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser is configured to operate in a content-loading mode and a content-loaded mode;	<p>"FIG. 2 illustrates a typical hardware configuration of a client 200 comprising a central processing unit (CPU) 210 coupled between a memory 214 and input/output (I/O) circuitry 218 by bidirectional buses 212 and 216." col. 8, ll. 14-15.</p> <p>"The other popular information service on the Internet is the Web. Instead of providing a user with a hierarchical list-oriented view of information, the Web provides the user with a 'linked-hypertext' view. Metaphorically, the Web perceives the Internet as a vast book of pages, each of which may contain pictures, text, sound, movies or various other types of data in the form of documents. Web documents are written in HyperText Markup Language (HTML) and Web servers transfer HTML documents to each other through the HyperText Transfer Protocol (HTTP)." col. 2, l. 60 - col. 3, l. 2.</p> <p>"As an example, a user 'traverses' the Web by following hot items of a page displayed on a graphical Web browser. These hot items are hypertext links whose presence are indicated on the page by visual cues, e.g., underlined words, icons or buttons. When a user follows a link (usually by clicking on the cue with a mouse), the browser displays the target pointed to by the link which, in some cases, may be another HTML document." col. 3, ll. 13-20.</p> <p>"A conventional display monitor 232 having a display screen 235 is also connected to I/O circuitry 218 via cable 238. A pointer (cursor) 240 is displayed on windows 244 of the screen 235 and its position is controllable via the mouse 230 or the keyboard 224, as is well-known." col. 8, ll. 31-36.</p>

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p>
		<p>"In accordance with the present invention, a novel network-oriented component layer 450 contains the underlying technology for implementing the extensible and replaceable network component system having an opening part component 475 that acts as a placeholder and displays progress during an asynchronous-event when navigating computer networks, such as the Internet." col. 9, ll. 27-33.</p> <p>"Referring again to FIG. 6, the window object 630 and the graphic interface object 650 are elements of a graphical user interface of a network component system having a customizable framework for greatly enhancing the ability of a user to monitor and display progress and status information during the opening process over a computer network." col. 14, ll. 53-58.</p> <p>"A CyberStream 710, which is a subclass that encapsulates the protocol knowledge required to download data referenced by objects of its companion CyberItem subclass, is used to provide progress information to a CyberDisplay part 720 and polls the CyberDisplay part 720 for progress information. For example, the progress information may include process mode data, process status data, and data on the percentage of the process that has been completed. A progress broadcaster object 730 and a progress receiver object 740 work in conjunction with a progress part 750 to monitor and display the progress information to a user. For instance, the progress broadcaster object 730 represents an asynchronous process, such as a download operation, and monitors the progress of that operation towards completion." col. 15, l. 55 - col. 16, l. 2.</p> <p>"Typical examples of monitored work include the functions necessary to complete downloading and e-mailing operations. For instance, in downloading operations, the progress broadcaster object 730 may monitor the number of bytes that have been downloaded to a disk and the total number of bytes that are expected to be downloaded." col. 16, ll. 8-14.</p> <p>"Referring to FIG. 8 for example, the progress part 750 may display progress information as a bar graph 752 and/or a numerical display 754. The progress part 750 obtains the status and progress information from its associated progress receiver 740</p>

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p> <p>for displaying this information on the GUI to the user." col. 16, ll. 26-31.</p> <p>"As indicated in step S150, a progress broadcaster is attached to the transient opener part for displaying progress and status information for the referenced data during this Resolve method. This transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." col. 16, ll. 54-60.</p>
40.4	<p>in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no "load status" graphic element is displayed, wherein absence of such "load status" graphic element indicates that the browser is in the content-loaded mode;</p>	<p>See FIG. 8 and the discussion that it may open in its own window: "As indicated in step S150, a progress broadcaster is attached to the transient opener part for displaying progress and status information for the referenced data during this Resolve method. This transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." col. 16, ll. 54-60.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to, in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no "load status" graphic element is displayed, wherein absence of such "load status" graphic element indicates that the browser is in the content-loaded mode.</p> <p>In addition, Nordman '850 identifies U.S. patent application Ser. No. 08/806,483 (now Cleron et al. U.S. Patent No. 5,907,843 ("Cleron '843")), also filed February 27, 1997, as a related copending U.S. application. col. 1, ll. 32-36. Cleron '843 states the following:</p> <p>"FIG. 8(B) illustrates an example of the navigator display part after a process has been completed. When a process has been completed and a page of information is being displayed, the status and progress banner 750, the process display 752 and the stop button 754 are eliminated from the display because they no longer provide any useful information. However, once another process is initiated, these displays and functions will be provided." col. 16, ll. 49-56.</p> <p>Thus, the combination of Nordman '850 and Cleron '843 renders this claim obvious.</p>

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p> <p>Nordman '850 and Cleron '843 teach, in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no "load status" graphic element is displayed, wherein absence of such "load status" graphic element indicates that the browser is in the content-loaded mode. One of skill in the art would be motivated to combine these references because they are in the same field and share common subject matter, address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
40.5	<p>in the content-loading mode, the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a "load status" graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such "load status" graphic element indicates that the browser is in the content-loading mode; and</p>	<p>"In accordance with the present invention, a novel network-oriented component layer 450 contains the underlying technology for implementing the extensible and replaceable network component system having an opening part component 475 that acts as a placeholder and displays progress during an asynchronous-event when navigating computer networks, such as the Internet." col. 9, ll. 27-33.</p> <p>"Referring again to FIG. 6, the window object 630 and the graphic interface object 650 are elements of a graphical user interface of a network component system having a customizable framework for greatly enhancing the ability of a user to monitor and display progress and status information during the opening process over a computer network." col. 14, ll. 53-58.</p> <p>"A CyberStream 710, which is a subclass that encapsulates the protocol knowledge required to download data referenced by objects of its companion CyberItem subclass, is used to provide progress information to a CyberDisplay part 720 and polls the CyberDisplay part 720 for progress information. For example, the progress information may include process mode data, process status data, and data on the percentage of the process that has been completed. A progress broadcaster object 730 and a progress receiver object 740 work in conjunction with a progress part 750 to monitor and display the progress information to a user. For instance, the progress broadcaster object 730 represents an asynchronous process, such as a download</p>

Claim	U.S. Patent 6,339,780	<p>Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348")² and/or PenPoint³ and/or various systems implementing a web browser on a mobile device</p>
		<p>operation, and monitors the progress of that operation towards completion." col. 15, l. 55 - col. 16, l. 2.</p> <p>"Typical examples of monitored work include the functions necessary to complete downloading and e-mailing operations. For instance, in downloading operations, the progress broadcaster object 730 may monitor the number of bytes that have been downloaded to a disk and the total number of bytes that are expected to be downloaded." col. 16, ll. 8-14.</p> <p>"Referring to FIG. 8 for example, the progress part 750 may display progress information as a bar graph 752 and/or a numerical display 754. The progress part 750 obtains the status and progress information from its associated progress receiver 740 for displaying this information on the GUI to the user." col. 16, ll. 26-31.</p> <p>"As indicated in step S150, a progress broadcaster is attached to the transient opener part for displaying progress and status information for the referenced data during this Resolve method. This transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." col. 16, ll. 54-60.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a "load status" graphic element over the content view area obstructing part of the content displayed in the content viewing area. Nordman '850 discloses that "[t]his transient opener part opens in its own window or in an embedded frame for displaying progress during the transient opening process of the Resolve method." e.g., a "load status" graphic element over the content view area obstructing part of the content displayed in the content viewing area. Based on this, one of skill in the art would have been motivated to implement a browser display a "load status" graphic element over the content view area obstructing part of the content displayed in the content viewing area.</p> <p>In addition, the combination of Nordman '850 and/or Vaughton '744 and/or Alley '282 and/or Jaaskelainen '348 and/or PenPoint renders this claim obvious. Vaughton</p>

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
		'744 and/or Alley '282 and/or Jaaskelainen '348 and/or PenPoint teaches displaying a "load status" graphic element over the content view area obstructing part of the content displayed in the content viewing area (see analysis of Vaughton '744 and/or Alley '282 in claim chart; see also Jaaskelainen '348 at FIG. 1 and PenPoint at 64-65 (describing the placement of the animated busy clock)) and it would be obvious to one of skill in the art to modify Nordman '850 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40.6	wherein content comprises data for presentation which is from a source external to the browser.	<i>See claim 32.5.</i>
41	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<i>See claim 10.</i>
42	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a	<i>See claim 11.</i>

Claim	U.S. Patent 6,339,780	Nordman '850 alone and/or in combination with U.S. Patent 5,907,843 ("Cleron '843") and/or U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 5,845,282 ("Alley '282") and/or U.S. Patent 5,301,348 ("Jaaskelainen '348") ² and/or PenPoint ³ and/or various systems implementing a web browser on a mobile device
	group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	

EXHIBIT P

Exemplar Chart of U.S. Patent 6,339,780**U.S. Patent 5,715,416 (“Baker ‘416’”)¹
Claims 1-6, 9-14, 17, 18, 20, 21 and 32-42**

Claim	U.S. Patent 6,339,780	Baker ‘416 alone and/or in combination with U.S. Patent 5,572,643 (“Judson ‘643’”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698’”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978’”) and/or U.S. Patent 5,864,850 (“Nordman ‘850’”) and/or various systems implementing a web browser on a mobile device
1	A hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area,	<p>“Navigating a well designed hypertext application can give the impression of walking around in the real world.” col. 7, ll. 35-37.</p> <p>“Hypermedia applications that make use of a worlds metaphor appear to be particularly appealing to children.” col. 7, ll. 55-56.</p> <p>“With the exception of the alternative desktop programs such as KidDesk, described earlier, hypermedia software applications are not intended to be used for general access to the file system and computer operating system. Users traverse the multimedia world designed by application developers by following hypermedia links or by moving a mouse pointer through a predefined three dimensional model.” col. 8, ll. 1-8.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser on an information processing device having</p>

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “User Definable Pictorial Interface for a Accessing Information in an Electronic File System” issued to Baker; filed September 30, 1994; issued February 3, 1998.

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
		<p>a limited display area.</p> <p>In addition, the combination of Baker '416 and one or more references describing handheld devices with browsers renders this claim obvious. Gessler², Cooper³, Bartlett⁴, Watson⁵, Kamba⁶, and Lauff⁷ each teach a hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area (e.g., a PDA, handheld, mobile device, etc.), and it would be obvious to one of skill in the art to modify Baker '416 with these teachings.⁸ One of skill in the art would be motivated to combine these references because they</p>

² S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. Computer Networks and ISDN Systems 28 (1995) 53-59 ("Gessler") (see pgs. 53 and 56-58).

³ I. Cooper and R. Shufflebotham. PDAWeb Browsers: Implementation Issues. (University of Kent at Canterbury, November 9, 1995) ("Cooper") (see pgs. 3 and 5).

⁴ J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE COMPCON 95 (San Francisco, March 1995) ("Bartlett") (see pgs. 1, 4 and 5).

⁵ T. Watson. "Application Design for Wireless Computing" (1994 Mobile Computing Systems and Applications Workshop Position Paper (August 1994)).

⁶ T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). "Using small screen space more efficiently."

⁷ M. Lauff and W. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.

⁸ Motorola continues to investigate other browsers that were in existence prior to the priority date of the '780 patent (e.g., NetHopper, Newt's Cape, PocketWeb and Vagon).

Claim	U.S. Patent 6,339,780	<p>Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device</p> <p>are all in the same field and share common subject matter (systems for downloading files from another system). Additionally, Bartlett cites Gessler and Cooper cites Gessler and Bartlett. Moreover, all of these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.1	<p>wherein the hypermedia browser has a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading content,</p>	<p><i>See</i> FIGS. 1, 1a, 3, 3a, 3b, and 3c.</p> <p>"It is believed that animations can be used to improve the expressiveness and extend the amount of information that can be conveyed in an icon. Some of this research has been incorporated into existing operating system interfaces, particularly for generic process depiction. For example, when an operation on a file is performed or a program is opened, the mouse pointer may become momentarily animated or may assume a different graphic, e.g. by displaying an hourglass." col. 3, ll. 45-53.</p> <p>"There has recently been a great deal of research focused on improving the ability of users to organize, browse, and retrieve files from very large file systems." col. 6, ll. 14-16.</p> <p>"In accord with these objects which will be discussed in detail below, the pictorial user interface of the present invention includes a pictorial image which is linked to a file directory and which identifies the file directory. Objects in the pictorial image are icons linked to file objects and an animated character is overlaid on the pictorial image. User input causes movement of the animated character relative to the pictorial image and animates objects in the pictorial image." col. 10, ll. 21-29.</p> <p>"An animated character 22 is overlaid on the background image 12 and is made responsive to an input device (shown and described with reference to FIGS. 2 and 2a) so that a user can control the movements and actions of the animated character 22." col. 13, ll. 5-9.</p>

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
		"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.
1.2	wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area,	See animated character 22 in FIGS. 1, 1a, 3, 3a, 3b, and 3c.
1.3	wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.	<p>Baker '416 discloses navigating and accessing complex file systems, including information stored on hard disks, CD-ROMs and networked file systems (e.g., accessing data for presentation from a source external to the browser). Baker '416 also discloses an animated character that roughly corresponds to the cursor or pointer in a WIMP GUI.</p> <p>"Because the capacity of storage devices such as hard disks and CD-ROMs is increasing and networked file systems are becoming prevalent, existing interfaces for file management are not able to effectively aid users attempting to manage or browse the enormous numbers of files now available to them. . . . There has recently been a great deal of research focused on improving the ability of users to organize." col. 6, ll. 1-16.</p> <p>"Recent advancements in computer graphics outside the realm of operating systems have enabled the development of highly intuitive application programs-particularly in the areas of education and entertainment. Much of this technology has been given the epithet of "multimedia" because it combines high resolution graphics, animation, video, and sound as well as ordinary text. There are now a large number of software</p>

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
		<p>application programs that use multimedia to create the impression of a complex "world" that can be traversed. In these applications, the user is presented with screens of graphical information. Each screen can have several hot spots that behave like operating system icons in that they respond to pointing and clicking of the mouse pointer. Typical actions in response to clicking on a hot spot include: displaying another screen of graphics, playing a sound, displaying an animation, displaying text, displaying a video, or a combination of these actions. Navigating a well designed hypertext application can give the impression of walking around in the real world. The user can look at where the user wants to go and go there by clicking the mouse pointer on an icon that points in that direction. The user can examine objects by pointing and clicking on them. The user can pick up objects, put them away, carry them, return to where the user started and go off in another direction. Some of these applications contain virtually no text at all and the user freely "walks" through thousands of graphic screens, views video clips and hears sounds along the way. For example, the user may enter through the door of a building by clicking the mouse pointer on the door and see many bookshelves inside. As the user approaches a bookshelf, by pointing the mouse pointer at the bookshelf and clicking the mouse button, titles of the books come into view. The user may select a book by clicking the mouse pointer on the spine of the book and the book will open showing the contents of the book. Pages are turned forward or back by clicking the pointer on the corner or edge of the page to be turned." col. 7, ll. 19-54.</p> <p>"The Pictorial Display. Turning now to FIGS. 1 and 1a, the pictorial user interface according to the invention provides a screen display 10 of a pictorial background image 12 (FIG. 1a) which represents a directory in a computer file system. For example, the picture displayed in FIG. 1 is of a roman nobleman and slave which represents a directory whose contents contains a draft of a book on Ancient Roman History. According to the invention, the pictorial information in the background image 12 is preferably metaphorical of the subject matter content of the directory. Moreover, individual sub-images 14, 16, and 18 (FIG. 1a.) (hereinafter referred to as "cons" for</p>

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
		simplicity) in the background image 12 relate to individual files or subdirectories contained in the directory represented by the background image 12. Each of these icons is also preferably metaphorical of the subject matter content of the file or subdirectory to which it relates. For example, the water urn icon 14 relates to a file about Ancient Roman ceramics and the nobleman icon 16 relates to a subdirectory containing information about the politics and government in the Roman Empire. At least one icon in the screen display relates to an ancestor directory unless the background currently displayed is the super root directory. For example, in FIG. 1a, the ladder icon 20 relates to the parent directory of the current directory. As seen in FIG. 1, the icons need not be rectangular and they may vary considerably in size. Although the invention attempts to obviate the need for textual information in the screen display, text may be included, at a user's option, within icons to further define their meaning. For example, in FIG. 1a, the stucco wall icon 18 includes the text string "Contents" to identify it as a file containing a Table of Contents to the book which is contained in the directory represented by the background image 12. An animated character 22 is overlaid on the background image 12 and is made responsive to an input device (shown and described with reference to FIGS. 2 and 2a) so that a user can control the movements and actions of the animated character 22. According to the invention, the animated character 22 may be moved to different positions on the screen display 10 to interact with any of the icons in the background image 12. The animated character roughly corresponds to the cursor or pointer in a WIMP GUI." col. 12, l. 38 - col. 13, l. 13.
2	A hypermedia browser as recited in claim 1, wherein the browser is configured to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.	According to Baker '416, different animations may relate to specific operating system actions. "When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user

Claim	U.S. Patent 6,339,780	<p>Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device</p> <p>specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content. For example, it would be obvious to employ a unique animation to represent the loading of data after selecting a new link.</p>
3	<p>A hypermedia browser as recited in claim 1, wherein the temporary graphic element indicates to a user that the browser is loading content.</p>	<p>According to Baker '416, different animations may relate to specific operating system actions.</p> <p>"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to employ the temporary graphic element to indicate to the user that the browser is loading content. For example, it would be obvious to employ a unique animation to represent the loading of data after selecting a new link.</p>
4	<p>A hypermedia browser as recited in claim 1, wherein the temporary graphic element disappears when the browser's loading of content is complete to indicate to a user that such loading of content is complete.</p>	<p>According to Baker '416, different animations may relate to specific operating system actions.</p> <p>"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user</p>

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
		specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to remove the temporary graphic element (e.g., stop or change the animation) when the browser's loading of content is complete to indicate to a user that such loading of content is complete. For example, it would be obvious to employ a unique animation to represent the loading of data after selecting a new link.
5	A hypermedia browser as recited in claim 1, wherein the temporary graphic element is animated.	"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.
6	A hypermedia browser as recited in claim 1, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	<i>See</i> FIGS. 3b and 3c.
9	A hypermedia browser as recited in claim 1, wherein the temporary graphic element conveys status information of the browser.	"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.
10	A hypermedia browser of claim 1,	"User definable hot spots that respond to mouse clicks are provided by hypermedia

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>authoring tools such as Apple's Hypercard, IBM's Linkway and AVC, Asymetrix' Toolbook, Macromedia Director, and others. Once defined, these hotspots behave like the icons in an operating system interface in that they respond to mouse clicks by executing an action. Unlike operating system icons, hot spots defined using a hypermedia authoring tool can be represented by any graphic and can be linked to any type of behavior." col. 8, ll. 36-44.</p> <p>"These programs are very powerful and give a skilled programmer the ability to create the sophisticated hypermedia applications described above such as Myst or Treehouse. Learning to use these tools to develop a hypermedia application generally takes many weeks or months and is therefore an activity normally carried out only by professionals or committed hobbyists. Moreover, it is generally necessary to make use of more specialized development tools to produce the graphics, sound, and animations required for a hypermedia application." col. 8, ll. 56-65.</p> <p>"The authoring tool simplifies the job of programming a multimedia hyperlinked application by giving a programmer ready made modules for multimedia such as animation and sound playback, and providing an interface that makes it easier to view, cut, and paste graphics and sound developed elsewhere, and to link the display of graphic scenes or execution of arbitrary actions to hotspots." col. 9, ll. 1-8.</p>
11	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>"User definable hot spots that respond to mouse clicks are provided by hypermedia authoring tools such as Apple's Hypercard, IBM's Linkway and AVC, Asymetrix' Toolbook, Macromedia Director, and others. Once defined, these hotspots behave like the icons in an operating system interface in that they respond to mouse clicks by executing an action. Unlike operating system icons, hot spots defined using a hypermedia authoring tool can be represented by any graphic and can be linked to any type of behavior." col. 8, ll. 36-44.</p> <p>"These programs are very powerful and give a skilled programmer the ability to create the sophisticated hypermedia applications described above such as Myst or Treehouse. Learning to use these tools to develop a hypermedia application generally takes many</p>

Claim	U.S. Patent 6,339,780		<p>Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device</p> <p>weeks or months and is therefore an activity normally carried out only by professionals or committed hobbyists. Moreover, it is generally necessary to make use of more specialized development tools to produce the graphics, sound, and animations required for a hypermedia application." col. 8, ll. 56-65.</p> <p>"The authoring tool simplifies the job of programming a multimedia hyperlinked application by giving a programmer ready made modules for multimedia such as animation and sound playback, and providing an interface that makes it easier to view, cut, and paste graphics and sound developed elsewhere, and to link the display of graphic scenes or execution of arbitrary actions to hotspots." col. 9, ll. 1-8.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement HyperCard wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web.</p> <p>In addition, the combination of HyperCard and Judson '643 and/or Lavey '698 and/or Nguyen '498 and/or Nguyen '978 renders this claim obvious. Judson '643 and/or Lavey '698 and/or Nguyen '498 and/or Nguyen '978 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web (see analysis of Judson '643 and/or Lavey '698 and/or Nguyen '498 and/or Nguyen '978 in claim chart) and it would be obvious to one of skill in the art to modify HyperCard with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure</p>
-------	-----------------------	--	---

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
		to do so, and there were a finite number of identified, predictable solutions.
12	An information processing device comprising:	"The invention relates to a graphical user interface for accessing information stored in a computer. More particularly, the invention relates to a user definable graphical interface for a computer operating system which utilizes pictorial information and animation as well as sound." col. 1, ll. 8-12.
12.1	a processor;	Baker '416 describes implementing its disclosed embodiments on a computer, which necessarily includes a processor. Therefore, one of ordinary skill in the art would readily recognize that Baker '416 inherently discloses a processor.
12.2	a display;	"This is another result of the fact that animations displayed by the operating system interface must be constructed in advance by software developers." col. 4, ll. 17-19.
12.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser has a content viewing area for viewing content and wherein the hypermedia browser displays a temporary graphic element over the content viewing area during times when the browser is loading visible content;	<i>See</i> claim 1.1.
12.4	wherein the temporary graphic element is positioned only over a portion of the content viewing area and obstructs only part of the visible content in the content viewing area; and	<i>See</i> claim 1.2.
12.5	wherein the temporary graphic element	According to Baker '416, different animations may relate to specific operating system

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	indicates to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser.	<p>actions.</p> <p>"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to employ the temporary graphic element to indicate to the user that the browser is loading content. For example, it would be obvious to employ a unique animation to represent the loading of data after selecting a new link.</p> <p>"Because the capacity of storage devices such as hard disks and CD-ROMs is increasing and networked file systems are becoming prevalent, existing interfaces for file management are not able to effectively aid users attempting to manage or browse the enormous numbers of files now available to them. . . . There has recently been a great deal of research focused on improving the ability of users to organize." col. 6, ll. 1-16.</p> <p>"Recent advancements in computer graphics outside the realm of operating systems have enabled the development of highly intuitive application programs-particularly in the areas of education and entertainment. Much of this technology has been given the epithet of "multimedia" because it combines high resolution graphics, animation, video, and sound as well as ordinary text. There are now a large number of software application programs that use multimedia to create the impression of a complex "world" that can be traversed. In these applications, the user is presented with screens of graphical information. Each screen can have several hot spots that behave like operating system icons in that they respond to pointing and clicking of the mouse pointer. Typical actions in response to clicking on a hot spot include: displaying</p>

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
		<p>another screen of graphics, playing a sound, displaying an animation, displaying text, displaying a video, or a combination of these actions. Navigating a well designed hypertext application can give the impression of walking around in the real world. The user can look at where the user wants to go and go there by clicking the mouse pointer on an icon that points in that direction. The user can examine objects by pointing and clicking on them. The user can pick up objects, put them away, carry them, return to where the user started and go off in another direction. Some of these applications contain virtually no text at all and the user freely "walks" through thousands of graphic screens, views video clips and hears sounds along the way. For example, the user may enter through the door of a building by clicking the mouse pointer on the door and see many bookshelves inside. As the user approaches a bookshelf, by pointing the mouse pointer at the bookshelf and clicking the mouse button, titles of the books come into view. The user may select a book by clicking the mouse pointer on the spine of the book and the book will open showing the contents of the book. Pages are turned forward or back by clicking the pointer on the corner or edge of the page to be turned." col. 7, ll. 19-54.</p> <p>"The Pictorial Display. Turning now to FIGS. 1 and 1a, the pictorial user interface according to the invention provides a screen display 10 of a pictorial background image 12 (FIG. 1a) which represents a directory in a computer file system. For example, the picture displayed in FIG. 1 is of a roman nobleman and slave which represents a directory whose contents contains a draft of a book on Ancient Roman History. According to the invention, the pictorial information in the background image 12 is preferably metaphorical of the subject matter content of the directory. Moreover, individual sub-images 14, 16, and 18 (FIG. 1a.) (hereinafter referred to as "cons" for simplicity) in the background image 12 relate to individual files or subdirectories contained in the directory represented by the background image 12. Each of these icons is also preferably metaphorical of the subject matter content of the file or subdirectory to which it relates. For example, the water urn icon 14 relates to a file about Ancient Roman ceramics and the nobleman icon 16 relates to a subdirectory</p>

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
		containing information about the politics and government in the Roman Empire. At least one icon in the screen display relates to an ancestor directory unless the background currently displayed is the super root directory. For example, in FIG. 1a. the ladder icon 20 relates to the parent directory of the current directory. As seen in FIG. 1, the icons need not be rectangular and they may vary considerably in size. Although the invention attempts to obviate the need for textual information in the screen display, text may be included, at a user's option, within icons to further define their meaning. For example, in FIG. 1a. the stucco wall icon 18 includes the text string "Contents" to identify it as a file containing a Table of Contents to the book which is contained in the directory represented by the background image 12. An animated character 22 is overlaid on the background image 12 and is made responsive to an input device (shown and described with reference to FIGS. 2 and 2a) so that a user can control the movements and actions of the animated character 22. According to the invention, the animated character 22 may be moved to different positions on the screen display 10 to interact with any of the icons in the background image 12. The animated character roughly corresponds to the cursor or pointer in a WIMP GUI." col. 12, l. 38 - col. 13, l. 13.
13	An information processing device as recited in claim 12, wherein the temporary graphic element is animated.	See claim 5.
14	An information processing device as recited in claim 12, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	See claim 6.
17	A hypermedia browser of claim 12, wherein content is data formatted for	See claim 10.

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	
18	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.
20	An information processing device as recited in claim 12, wherein the temporary graphic element is not content.	See claim 1.3.
21	An information processing device as recited in claim 12, wherein the temporary graphic element disappears when the browser's loading of content is complete to indicate to a user that such loading of content is complete.	See claim 4.
32	A method of indicating a content "load status" of a hypermedia browser having a content viewing area for viewing content,	"It is believed that animations can be used to improve the expressiveness and extend the amount of information that can be conveyed in an icon. Some of this research has been incorporated into existing operating system interfaces, particularly for generic

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	the method comprising:	process depiction. For example, when an operation on a file is performed or a program is opened, the mouse pointer may become momentarily animated or may assume a different graphic, e.g. by displaying an hourglass." col. 3, ll. 45-53.
32.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a "load status" graphic element, wherein a "load status" graphic element indicates a current content load status of the hypermedia browser;	<p>"It is believed that animations can be used to improve the expressiveness and extend the amount of information that can be conveyed in an icon. Some of this research has been incorporated into existing operating system interfaces, particularly for generic process depiction. For example, when an operation on a file is performed or a program is opened, the mouse pointer may become momentarily animated or may assume a different graphic, e.g. by displaying an hourglass." col. 3, ll. 45-53.</p> <p>"In accord with these objects which will be discussed in detail below, the pictorial user interface of the present invention includes a pictorial image which is linked to a file directory and which identifies the file directory. Objects in the pictorial image are icons linked to file objects and an animated character is overlaid on the pictorial image. User input causes movement of the animated character relative to the pictorial image and animates objects in the pictorial image." col. 10, ll. 21-29.</p> <p>"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to employ a "load status" graphic element to indicate a current content load status of the hypermedia browser. For example, it would be obvious to employ a unique animation to represent the loading of data after selecting a new link.</p>
32.2	receiving an instruction to load new	"With the exception of the alternative desktop programs such as KidDesk, described

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	content into the content viewing area;	earlier, hypermedia software applications are not intended to be used for general access to the file system and computer operating system. Users traverse the multimedia world designed by application developers by following hypermedia links or by moving a mouse pointer through a predefined three dimensional model." col. 8, ll. 1-8. "When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.
32.3	loading such new content into the content viewing area; and	"There are now a large number of software application programs that use multimedia to create the impression of a complex "world" that can be traversed. In these applications, the user is presented with screens of graphical information. Each screen can have several hot spots that behave like operating system icons in that they respond to pointing and clicking of the mouse pointer. Typical actions in response to clicking on a hot spot include: displaying another screen of graphics, playing a sound, displaying an animation, displaying text, displaying a video, or a combination of these actions. Navigating a well designed hypertext application can give the impression of walking around in the real world. The user can look at where the user wants to go and go there by clicking the mouse pointer on an icon that points in that direction." col. 7, ll. 25-39.
32.4	while loading, displaying a "load status" graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	See FIGS. 1, 1a, 3, 3a, 3b, and 3c. "It is believed that animations can be used to improve the expressiveness and extend the amount of information that can be conveyed in an icon. Some of this research has been incorporated into existing operating system interfaces, particularly for generic

Claim	U.S. Patent 6,339,780	<p>Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device</p> <p>process depiction. For example, when an operation on a file is performed or a program is opened, the mouse pointer may become momentarily animated or may assume a different graphic, e.g. by displaying an hourglass." col. 3, ll. 45-53.</p> <p>"There has recently been a great deal of research focused on improving the ability of users to organize, browse, and retrieve files from very large file systems." col. 6, ll. 14-16.</p> <p>"In accord with these objects which will be discussed in detail below, the pictorial user interface of the present invention includes a pictorial image which is linked to a file directory and which identifies the file directory. Objects in the pictorial image are icons linked to file objects and an animated character is overlaid on the pictorial image. User input causes movement of the animated character relative to the pictorial image and animates objects in the pictorial image." col. 10, ll. 21-29.</p> <p>"An animated character 22 is overlaid on the background image 12 and is made responsive to an input device (shown and described with reference to FIGS. 2 and 2a) so that a user can control the movements and actions of the animated character 22." col. 13, ll. 5-9.</p> <p>"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.</p>
32.5	wherein content comprises data for presentation which is from a source external to the browser.	See claim 1.3.
33	A method as recited in claim 32 further	According to Baker '416, different animations may relate to specific operating system

Claim	U.S. Patent 6,339,780	<p>Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device</p>
	<p>comprising, upon completion of the loading, removing the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.</p>	<p>actions.</p> <p>"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to, upon completion of the loading, remove the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed (e.g., stop, change or remove the animation). For example, it would be obvious to employ a unique animation to represent the loading of data after selecting a new link.</p>
34	<p>A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.</p>	<p>See claim 10.</p>
35	<p>A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X,</p>	<p>See claim 11.</p>

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	Flash. scripting language for the world wide web.	
36	A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method of indicating a content "load status" of a hypermedia browser having a content viewing area for viewing content, the method comprising:	<p>"The invention relates to a graphical user interface for accessing information stored in a computer. More particularly, the invention relates to a user definable graphical interface for a computer operating system which utilizes pictorial information and animation as well as sound." col. 1, ll. 8-12.</p> <p>"It is believed that animations can be used to improve the expressiveness and extend the amount of information that can be conveyed in an icon. Some of this research has been incorporated into existing operating system interfaces, particularly for generic process depiction. For example, when an operation on a file is performed or a program is opened, the mouse pointer may become momentarily animated or may assume a different graphic, e.g. by displaying an hourglass." col. 3, ll. 45-53.</p>
36.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen is without a "load status" graphic element, wherein a "load status" graphic element indicates a current content load status of the hypermedia browser;	See claim 32.1.
36.2	receiving an instruction to load new content into the content viewing area;	See claim 32.2.
36.3	loading such new content into the content viewing area; and	See claim 32.3.
36.4	while loading, displaying a "load status"	See claim 32.4.

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	
36.5	wherein content comprises data for presentation which is from a source external to the browser.	See claim 32.5.
37	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	See claim 10.
38	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	See claim 11.
39	A computer-readable medium as recited in claim 36 further having additional computer-executable instructions that	See claim 33.

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	perform a method comprising, upon completion of the loading, removing the "load status" graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	
40	An information processing device comprising:	See claim 12.
40.1	a processor;	See claim 12.1.
40.2	a display;	See claim 12.2.
40.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser is configured to operate in a content-loading mode and a content-loaded mode;	<p>"In accord with these objects which will be discussed in detail below, the pictorial user interface of the present invention includes a pictorial image which is linked to a file directory and which identifies the file directory. Objects in the pictorial image are icons linked to file objects and an animated character is overlaid on the pictorial image. User input causes movement of the animated character relative to the pictorial image and animates objects in the pictorial image." col. 10, ll. 21-29.</p> <p>"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58.</p>
40.4	in the content-loaded mode, the hypermedia browser displays loaded	"It is believed that animations can be used to improve the expressiveness and extend the amount of information that can be conveyed in an icon. Some of this research has

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	content in the content viewing area and no "load status" graphic element is displayed, wherein absence of such "load status" graphic element indicates that the browser is in the content-loaded mode;	<p>been incorporated into existing operating system interfaces, particularly for generic process depiction. For example, when an operation on a file is performed or a program is opened, the mouse pointer may become momentarily animated or may assume a different graphic, e.g. by displaying an hourglass." col. 3, ll. 45-53.</p> <p>"There are now a large number of software application programs that use multimedia to create the impression of a complex "world" that can be traversed. In these applications, the user is presented with screens of graphical information. Each screen can have several hot spots that behave like operating system icons in that they respond to pointing and clicking of the mouse pointer. Typical actions in response to clicking on a hot spot include: displaying another screen of graphics, playing a sound, displaying an animation, displaying text, displaying a video, or a combination of these actions. Navigating a well designed hypertext application can give the impression of walking around in the real world. The user can look at where the user wants to go and go there by clicking the mouse pointer on an icon that points in that direction." col. 7, ll. 25-39.</p> <p>"In accord with these objects which will be discussed in detail below, the pictorial user interface of the present invention includes a pictorial image which is linked to a file directory and which identifies the file directory. Objects in the pictorial image are icons linked to file objects and an animated character is overlaid on the pictorial image. User input causes movement of the animated character relative to the pictorial image and animates objects in the pictorial image." col. 10, ll. 21-29.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to, in the content-loaded mode, display loaded content in the content viewing area and no "load status" graphic element, wherein absence of such "load status" graphic element indicates that the browser is in the content-loaded mode. For example, it would be obvious to employ a unique animation to represent the loading of data after selecting a new link that is not displayed in the content-loaded mode.</p>
40.5	in the content-loading mode, the	"It is believed that animations can be used to improve the expressiveness and extend

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a "load status" graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such "load status" graphic element indicates that the browser is in the content-loading mode; and	<p>the amount of information that can be conveyed in an icon. Some of this research has been incorporated into existing operating system interfaces, particularly for generic process depiction. For example, when an operation on a file is performed or a program is opened, the mouse pointer may become momentarily animated or may assume a different graphic, e.g. by displaying an hourglass." col. 3, ll. 45-53.</p> <p>"There are now a large number of software application programs that use multimedia to create the impression of a complex "world" that can be traversed. In these applications, the user is presented with screens of graphical information. Each screen can have several hot spots that behave like operating system icons in that they respond to pointing and clicking of the mouse pointer. Typical actions in response to clicking on a hot spot include: displaying another screen of graphics, playing a sound, displaying an animation, displaying text, displaying a video, or a combination of these actions. Navigating a well designed hypertext application can give the impression of walking around in the real world. The user can look at where the user wants to go and go there by clicking the mouse pointer on an icon that points in that direction." col. 7, ll. 25-39.</p> <p>"In accord with these objects which will be discussed in detail below, the pictorial user interface of the present invention includes a pictorial image which is linked to a file directory and which identifies the file directory. Objects in the pictorial image are icons linked to file objects and an animated character is overlaid on the pictorial image. User input causes movement of the animated character relative to the pictorial image and animates objects in the pictorial image." col. 10, ll. 21-29.</p> <p>"An animated character 22 is overlaid on the background image 12 and is made responsive to an input device (shown and described with reference to FIGS. 2 and 2a) so that a user can control the movements and actions of the animated character 22." col. 13, ll. 5-9.</p> <p>"When the animated character 22 interacts with an icon, both the character 22 and the icon exhibit an animation. The animation exhibited is preferably metaphorical of</p>

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
		some operating system action. As will be described in detail below, the background images and sub-images are selectable and definable by the user to relate to user specified files and directories. In addition, the animated character and its animations are selectable and at least partially definable by the user to relate to specific operating system actions." col. 13, ll. 49-58. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to, in the content-loading mode, load content, display such content in the content viewing area as it loads, and display a "load status" graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such "load status" graphic element indicates that the browser is in the content-loading mode. For example, it would be obvious to employ a unique animation to represent the loading of data after selecting a new link that is only displayed in the content-loading mode.
40.6	wherein content comprises data for presentation which is from a source external to the browser.	See claim 32.5.
41	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	See claim 10.
42	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a	See claim 11.

Claim	U.S. Patent 6,339,780	Baker '416 alone and/or in combination with U.S. Patent 5,572,643 ("Judson '643") and/or Lavey U.S. Patent 6,023,698 ("Lavey '698") and/or U.S. Patent 6,584,498 ("Nguyen '498") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or U.S. Patent 5,864,850 ("Nordman '850") and/or various systems implementing a web browser on a mobile device
	group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	